# Dynamically Accepting and Scheduling Patients for Home Healthcare

by

Mustafa DEMIRBILEK

Dissertation

Submitted to the University of Warwick for the degree of

Doctor of Philosophy

Warwick Business School

May 2018

# Contents

# List of Figures

# List of Tables

# List of Algorithms

# Acknowledgements

I would like to thank my supervisor, Dr Juergen Branke, for his kind support and guidance throughout my research. His supervision and advice are of inestimable value for my study. His guidance helped me in all the time of research and writing of this thesis.

I also would like to thank my co-supervisor, Dr Arne Strauss, for his insightful comments and suggestions.

I would like to thank the Ministry of National Education, Republic of Turkey. I could not even start and pursue this great journey without their financial support.

Last but not the least, I would like to thank my family and friends for supporting me spiritually throughout writing this thesis and my life in general.

# Declarations

I hereby declare that this thesis, submitted to the University of Warwick in support of my application for the degree of Doctor of Philosophy, has been composed by myself and has not been submitted in any previous application for any degree. The work presented (including data generated and data analysis) was carried out by the author.

This study was presented in Operational Research Applied to Health Services Conference, 2017.
A part of this thesis was published by the author:
Demirbilek Mustafa, Juergen Branke, and Arne Strauss. "Dynamically accepting and scheduling patients for home healthcare." Health care management science (2018): 1-16.
Another part of this thesis was submitted to Flexible Services and Manufacturing Journal in April 2018 and has been under review since then.

Signature: ..............................................................

Date: ..............................................................

# Abstract

Importance of home healthcare is growing rapidly since populations of developed and even developing countries are getting older quickly and the number of hospitals, retirement homes, and medical staff do not increase at the same rate. We present Scenario Based Approach (SBA) for the Home Healthcare Nurse Scheduling Problem. In this problem, arrivals of patients are dynamic and acceptance and appointment time decisions have to be made as soon as patients arrive. The primary objective is to maximise the average number of daily visits. For the sake of service continuity, patients have to be visited at the same days and times each week during their service horizon. SBA is basically a simulation procedure based on generating several scenarios and scheduling new customers with a simple but fast heuristic. Then results are analysed to decide whether to accept the new patient and at which appointment day/time. First, two different versions of SBA, Daily and Weekly SBA are developed and analysed for a single nurse. We compare Daily SBA to two greedy heuristics from the literature, distance and capacity based, and computational studies show that Daily SBA makes significant improvements compared to these other two methods for a single nurse. Next, we extend SBA for a multi-nurse case. SBA is compared to a greedy heuristic under different conditions such as same depot case where nurses start their visits from and return to same place, clustered service area, and nurses with different qualification level. SBA gives superior results under all experiment conditions compared to the greedy heuristic.

# Abbreviations

**CH** Capacity Heuristic

**CIH** Cheapest Insertion Heuristic

**DH** Distance Heuristic

**DHM** Distance Heuristic for Multiple Nurses

**DPVRP** Dynamic Periodic Vehicle Routing Problem

**DSBA** Daily Scenario Based Approach

**DVRP** Dynamic Vehicle Routing Problem

**GA** Genetic Algorithm

**HHC** Home Healthcare

**MILP** Mixed Integer Linear Programming

**NSP** Nurse Rostering Problem

**PSO** Particle Swarm Optimisation

**SA** Simulated Annealing

**SBA** Scenario Based Approach

**SBAM** Scenario Based Approach for Multiple Nurses

**TS** Tabu Search

**TSPTW** Traveling Salesman Problem with Time Windows

**VNS** Variable Neighbourhood Search

**VRP** Vehicle Routing Problem

**WSBA** Weekly Scenario Based Approach

# Chapter 1

# Introduction

Home healthcare (HHC), also referred to as in-home care, social care, or domiciliary care, is becoming one of the most important components of health care. HHC helps hospitals and retirement homes to create free capacity and decrease care delivering cost [Hall, 2012]. The most crucial objective of HHC is to ensure people who need medical attention and daily care to receive high-standard home services. According to patients' needs, nurses, physicians, doctors, and operators visit patients' homes periodically and provide services. Many elderly, chronically ill, and disabled people receive HHC services [CMS, 2008]. Although home care and HHC services refer to the same activity in the literature, they are different. On the one hand, home care includes daily activities such as cleaning, dressing, bathing, and cooking to help the elderly, on the other hand, HHC includes medical activities such as providing pills and shots, physical and mental rehabilitation, watching the daily medication regime. However, companies often provide both, home care and HHC, by employing trained and educated staff according to job's requirement.

In 2008, the US saved $25 billion in hospital costs thanks to HHC services according to the National Association for Home Care and Hospice [NAHC, 2016]. HHC companies employed 1.8 million caregivers and it was estimated that 500,000 more jobs were potentially created in 2014 [NAHC, 2016]. 40% of adults aged 65+ already take HHC service. The majority of HHC users are people with an average age of

69 [NAHC, 2016]. 59% of them have long-term physical conditions and 26% have memory problems. 70% of all Americans aged 65 and older will need HHC service at some points in their lives. Due to some factors such as aging population, chronic diseases, insufficient capacity of hospitals, etc., it was projected that the demand for HHC doubled by 2030 compared to 2010 [Albuquerque, 2010]. The following information shows why HHC is gaining much more importance day by day in the US:

- The number of people aged 65 and over in the US will be 56 million by 2020, numbers reach 84 million by 2040 [Census, 2011].
- Care of a patient in the home costs only $45,000 per year for average of 44 hours of care per week while $91,250 are spent for a patient receiving care in a nursing home [NAHC, 2016].
- Home-based health technologies cost $3 billion in 2007 versus $7.7 billion in 2012 [Hall, 2012].
- The percentage of American adults who are chronically ill is more than 50% [AHRQ, 2007].

We encounter similar situations in Europe as well. For instance, in Sweden, the total care cost was approximately €8.8 billion in 2005, and care of an elderly person cost €49,500 in a retirement home annually whereas only €20,300 was spent per person receiving care in their home. Moreover, roughly 88,000 of a total 250,000 staff were employed full-time in both public and private HHC organizations, which was 2% of Sweden's total work force [Eveborn et al., 2006]. In France, the total number of HHC providers increased from 68 in 1999 to 123 in 2005 and to 231 in 2008. The number of hours spent for HHC activities rose by 84% while the number of patients increased by 147% between 2005 and 2008 [Benzarti, 2012].

On the other side of the world, China is experiencing a huge demand of aging population for HHC services. The number of people aged over 60 in China was around 212 million, accounting for 15.5% of China's total population in 2015,

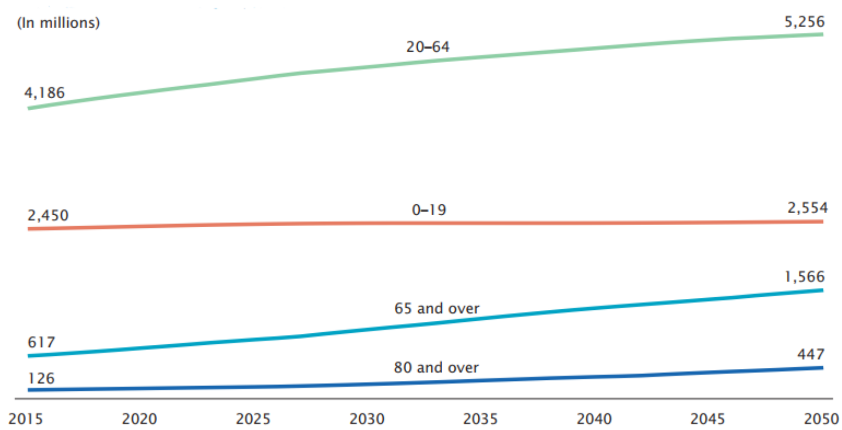



Figure 1.1: Projected change in the total number of people in different age groups in the world between 2015 and 2050 [NIH, 2015]

which was higher than the traditional standard aging society (10%) [Du et al., 2017]. Moreover, 80%-90% of the seniors have chronic diseases and need continuous health services [Du et al., 2017].

There are some factors that increase demand for HHC in the world. First, the rise of life expectancy causes demographic changes especially in developed countries. The proportion of elderly people has been going up for last several decades and is projected to rise significantly in all over the world as shown in Figure 1.1 and 1.2. Next, the number of people suffering from Alzheimer's and dementia or chronic diseases significantly rises, for example, the number of people with dementia doubled every 20 years [International, 2015]. HHC is very suitable for treatment of this kind of diseases. Another factor is that people who take HHC services do not need to leave from their homes, families, and their social life and this makes HHC more preferable than institutionalized care where people must stay as long as their treatment continues. Finally, HHC services are supported by governments thanks to their social and financial benefits [Benzarti, 2012].

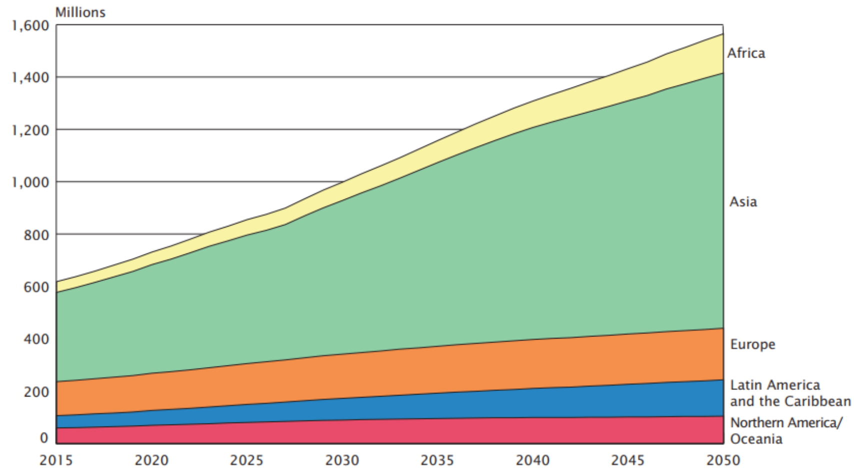



Figure 1.2: Population aged 65 and over by region: 2015 to 2050 [NIH, 2015]

## 1.1 The HHC Problem and Motivation

The HHC problem starts with a hospital request. When a hospital discharges a patient who still needs medical attention for a while, the service provider is informed what kind of treatment the patient needs and how many times he or she needs to be visited weekly. After that, the service provider has to decide when weekly visits take place during the service horizon of the patient. Furthermore, which nurse is assigned to visits should be determined according to qualifications, preferences, and availability of nurses. After constructing schedules, nurses start their daily trips from their homes, visit and service patients at pre-specified times, and return to their homes at the end of each day.

Although our problem setting based on the US HHC system [Bennett and Erera, 2011] is applicable for home care problem in where people and their relatives can apply directly, it is more suitable for patients being discharged from hospitals and whose needs (how many times weekly and how long they have to be visited) are already known since we do not consider extra time or maybe an extra visit for a triage in this study.

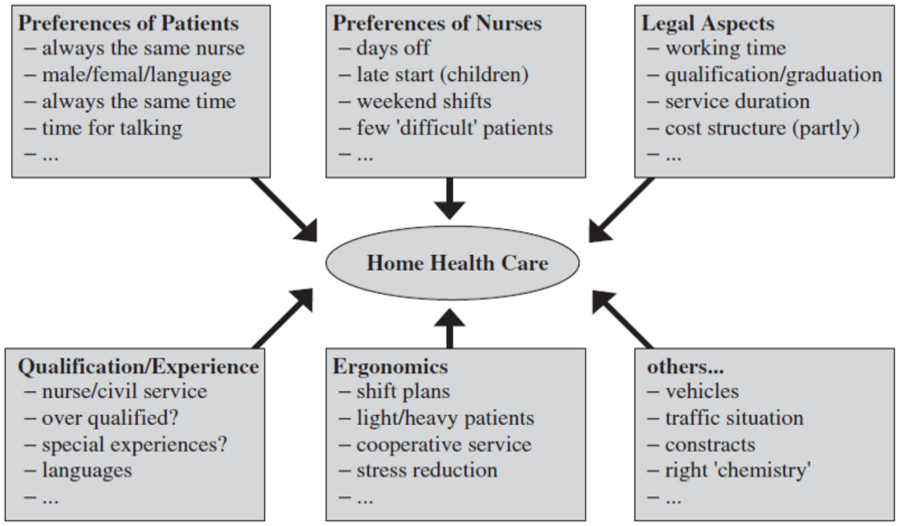



Figure 1.3: Some criteria and restrictions that affect the decision making process in HHC [Bertels and Fahle, 2006]

Although HHC has many different constraints and decision criteria as in Figure 1.3, routing and scheduling decisions are mostly made manually by a senior nurse or manager in many companies as can be seen in our literature review. Therefore, the proposed solution methodologies make great improvements in terms of daily travel times, daily visits, and patient/worker satisfactions compared to manual routing and scheduling. However, existing studies summarised in Table 2.3 assume that all patient requests are already known at the beginning of service horizon. Unfortunately, we do not find any explanation how all requests can be known in advance. One reason can be that requests are being collected till the beginning of a new planning term, which can be a week or a month. One question is whether or not people are willing to wait for a decision even though it is possible that their requests are not accepted due to scarce resource. Furthermore, delaying discharge of a patient from a hospital due to decision processes of HHC companies causes an extra cost for the hospital and dissatisfaction for the patient. The other question is how recently arrived patient requests are integrated into the existing schedule under long HHC service horizon.

In this work, we focus on dynamic patient arrivals and decisions where acceptance and assignment time decisions have to be made as soon as patients arrive. The most important challenge in the dynamic problem setting is how to decide visit days and times for a patient without knowing future patient requests. One way for a solution is to assign the patient the best days and times into the current schedule by simply ignoring future patients. Of course, this is not a good way since the best assignment we make now can be the worst depending on locations of future patients. Therefore, we consider future requests when assigning visits of current requests now in this study.

The other issue raised from this problem is whether or not accepting all requests is a good strategy. In the literature, an acceptance policy is also referred to as *service guarantee* and occasionally discussed in different areas such as public transportation [Li et al., 2009] and vehicle dispatching [Ichoua et al., 2000]. It is worth to investigate whether or not rejecting a patient located at an unsuitable place for the route of a nurse allows to accept more patient visits in the future. In this study, we do not only identify suitable assignment days and times, but also make accepting or rejecting decision for each patient.

## 1.2 Research Questions and Objectives

The fundamental research question pertaining to the scope of this study is:

*"How can we decide to accept or reject a patient, and if he or she is accepted, how to find suitable visit days and times by considering future demand?"*

Our objective is to maximise average daily visits during a service horizon. Furthermore, we explicitly consider travel times per visit, balance nurses' workloads, and acceptance rates.

## 1.3 Contributions

Our main contribution is twofold. First, we develop a new solution methodology for a HHC scheduling problem under dynamic patient arrivals. The new methodology finds the most suitable nurse, visit days, and times for each accepted patient depending on future demand to maximise daily visits. Second contribution is empirical. We test our algorithm under different problem settings such as service areas with different sizes and demand volumes, different weekly visits, service horizons and durations, continuity and non-continuity of services and so on. In Chapter 3, only a single nurse servicing patients in a specific area is considered and any overlaps with other nurses' regions are ignored. In particular, contributions include:

- A new acceptance and scheduling policy based on a solution methodology which anticipates future demand for the Dynamic HHC problem.
- A comparison of two different approaches, one depending on constructing tours for each day of the week independently and the other considering all visits of requests in the week at the same time when constructing tours for each day.
- A comparison of our solution method to two greedy heuristics proposed by [Bennett and Erera, 2011].
- Tests our algorithm under violation of the service continuity in terms of service times.
- A new pricing policy based on patient preferred visit days and times.

In Chapter 4, Scenario Based Approach (SBA) is modified to be able to consider more than one nurse. The modification is to take all nurses and weekly visits of patients into consideration at the same time. The main contributions can be defined as following:

- An improved algorithm that captures real life aspects with multiple nurses and different skill levels by anticipating future demand for the Dynamic HHC problem.

- An empirical insight how much better it is to plan nurses' routing and scheduling without restricting nurses to districts.
- Insights to algorithmic performance under different conditions such as clustered service areas, different service times and service horizons.
- Empirically demonstrated improvement over a benchmark heuristic proposed by [Bennett and Erera, 2011].
- Tests our algorithm under violation of the service continuity in terms of service times and nurses.
- A new pricing policy based on patient preferred visit days, times, and nurses.

As far as the literature relating to this research are concerned, the relevant publications are listed below:

- This study, titled "Dynamically accepting and scheduling patients for home healthcare.", was presented in Operational Research Applied to Health Services Conference, 2017.
- A journal paper, "Demirbilek Mustafa, Juergen Branke, and Arne Strauss. 'Dynamically accepting and scheduling patients for home healthcare.' Health care management science (2018): 1-16.", was published.
- A journal paper, "Demirbilek Mustafa, Juergen Branke, and Arne Strauss. 'Home Healthcare Routing and Scheduling of Multiple Nurses in a Dynamic Environment.'", was submitted to Flexible Services and Manufacturing Journal in April 2018 and has been under review since then.

## 1.4 Thesis Organisation

This thesis is composed of five chapters. The organisation of the thesis is as follows: In Chapter 2, we present a literature review related to home health nurse routing and scheduling problems as well as Dynamic Vehicle Routing and Periodic Dynamic Vehicle Routing Problems. In Chapter 3, we formally define the single nurse problem and present a solution methodology for a single nurse case. In Chapter 4, we extend our solution methodology for multiple nurses and examine the nurse districting problem and qualification levels. We conclude our study and talk about future opportunities in Chapter 5.

# Chapter 2

# Literature Review

In this section, we go over the most relevant studies in terms of the problem; nurse routing and scheduling problems, solution methodology; Dynamic Vehicle Routing Problem (DVRP)/Dynamic Periodic VRP (DVRP) studies, and others; nurse districting and HHC supply chain problems, due to the importance of nurse routing and scheduling models for our study. Although many opportunities exist to use operations research methods due to the complexity of HHC problems, very few papers exist in the literature. However, the number of publication has increased significantly since 2014 as shown in Figure 2.1.

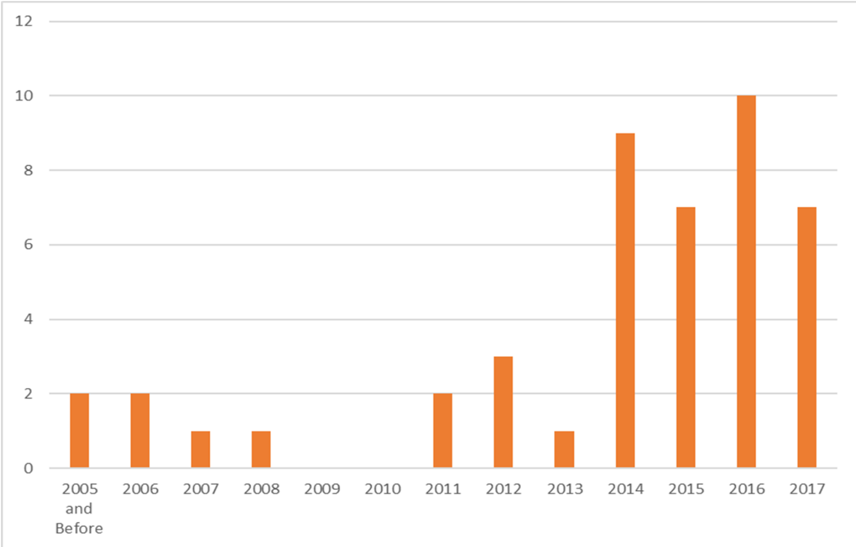



Figure 2.1: Distribution of HHC Nurse Routing and Scheduling studies in our literature review over time

## 2.1 HHC Nurse Routing and Scheduling Models

HHC related models started with [Fernandez et al., 1974], "A model for community nursing in a rural country.". They divided a whole service region into several subregions and assigned nurse teams to each subregions to be able to effectively use the limited number of nurses. The next important study that HHC related studies were getting more attentions after was "An Integrated Spatial DSS for Scheduling and Routing Home Health Care Nurses." [Begur et al., 1997] constructed a decision support system for a home care company to optimise their routing and rostering operation without considering time windows. Beside our comprehensive literature review, we also refer readers to [Fikar and Hirsch, 2017] and [Mutingi and Mbohwa, 2013] for a state-of-the-art review of the models and algorithms that have been reported in the HHC routing and scheduling literature between 1997 and 2016. Tables 2.1, 2.2, and 2.3 represent a classification of publications in terms of objectives/performance measures, constraints, and solution methodologies in the literature.

We divide existing studies into two main categories, Static and Dynamic Problems, since our main contributions are under consideration of dynamic problem settings. However, mentioning static problems and solutions are important to be able to show why we consider dynamism and develop a new solution method.

### 2.1.1 Static Problems

In static problems, all data are known in advance before the optimisation has started. Although studies can be categorised according to many criteria such as their objectives, constraints, and optimisation periods (single or multiple), we categorise studies in this section according to their solution methodologies. Many studies are carried out by HHC companies from different countries. Therefore, objectives, constraints, and periods vary based on requirements and work regulations of companies and countries. Thus, we focus on solution methodologies and divide studies into three categories, Exact Solution Methods, Heuristic/Metaheuristic Solution Methods, and Comparative Solutions.

#### 2.1.1.1 Exact Solution Methods

We can find relatively few publications that propose only exact solution methods for the HHC problem since the problem is a combination of two well known NP-hard problems, VRP and Nurse Rostering Problem (NSP) [Steeg and Schroder, 2008]. Exact solution methods works for only small instances. For example, an instance with up to 50 patients requires about 3 hours run time [Fikar and Hirsch, 2017]. Therefore, the following publications worked on small instances and daily or weekly routing and scheduling activities.

[Carello and Lanzarone, 2014] developed a healthcare application based on nurse rostering, taking into account the continuity of care requirement. They used the cardinality-constrained approach which exploited potentialities of a linear programming model, but avoided to generate scenarios for stochastic problem settings. They tested the approach by using real-life data, taken from a HHC service provider in

Italy, and observed that it showed superior results in terms of overtime work and continuity of care compared to non-robust algorithms. However, the algorithms could provide reasonable results for most a week since computational cost became very high for longer periods.

[Cappanera and Scutellà, 2014] tried to develop a model that took into account operators' skill level matching to patients' needs. They proposed an integer linear programming formulation to solve this assignment problem including scheduling and routing factors. They used real data derived from HHC providers in Italy to evaluate their model and observed that it worked successfully.

[Yalçındağ et al., 2016a] developed a two-phase solution methodology based on a similar study of [Cappanera and Scutellà, 2014]. The main difference between the two studies was that Yalcindag et al. decomposed the joint approach of Cappanera and Scutella that included assignment, scheduling, and routing solutions in order to solve large instances in a computationally more efficient way. They tested several two-phase combinations in which each phase can cover one or two of assignment, scheduling, and routing solutions. According to computational results based on large real world instances considering qualification levels, continuity of care, and multi-period planning horizon, the two-phase method (assignment and scheduling phase 1; routing in phase 2) provided computationally efficient results in terms of the optimality gap compared to the single-phase method by Cappanera and Scutella.

[Manerba and Mansini, 2016] proposed a HHC problem defined as an extension of the multi-vehicle travelling purchaser problem. They aimed to maximise the total benefit of performed services. The benefit of a service depended on importance and priority of patient visits. Incompatible services that cannot be performed for the same patient at the same day, qualification matching, and time windows were considered as constraints. They modelled the problem by mixed integer programming and proposed a branch and price approach as the solution method.

[Wirnitzer et al., 2016] developed a nurse rostering model for a HHC company to optimise scheduling activities done manually before. They proposed five mixed integer programming formulations. Each one had a different objective function targeting continuity of care from the perspective of patients with same hard constraints such as breaks, maximum daily and weekly working times, patient/nurse preferences, and shift rotations. They compared the results in terms of the number of assigned and switched nurses. According to results based on randomly generated data derived from real-world input and the company's data, all models outperformed the manual planning in reasonable time.

[Yalcindag et al., 2012] presented a two-stage approach for routing and rostering decisions for HHC organizations. Their model used the results of rostering problem as input to the routing problem. Specifically, they investigated whether or not different rostering models had an impact on the routing models. A mixed integer programming was employed to solve the rostering problem, considering workload balance and continuity of care. On the other hand, they formulated a travelling salesman problem model to solve the routing part to find the tour with minimum length.

[Issabakhsh et al., 2018] presented a robust mathematical model for patients who need peritoneal dialysis in their home. According to their model, patients had different requirements such as collection of urine or blood samples, visits by nurses and technicians, and deliveries of some necessary medicines etc. Because of such necessities, they had to take into account not only depots and patient's locations, but also dialysis centres and laboratories. By considering some constraints such as, labs had to be visited after collecting blood or urine tests, and nurses and technicians had to be taken from a dialysis centre before visits of patients. Since just on time visits were a important factor for peritoneal dialysis, they developed the robust optimisation model to handle uncertainty in travel times. It turned out that in even the most uncertain scenario, the differences between the robust and deterministic results were less than 1.2%.

[En-nahli et al., 2015] proposed a solution methodology for HHC by considering multiple objectives at the same time. They maximised caregivers utilizations and fairness of workloads while minimising travel times and waiting times of caregivers. Their model included service continuity, time windows, skill level compatibility, affinity between patients and caregivers, lunch breaks and pre-specified work times as constraints. To solve this problem, authors presented a mixed linear integer programming and ILOG Cplex Solver was employed.

[Masmoudi and Mellouli, 2014] considered a HHC problem as an application of a synchronized multiple Travelling Salesman Problem with Time Windows (TSPTW). The synchronisation in the problem referred to the condition that some patients needed more than one staff member at the same time. Their aim was to minimise the total travelling and waiting times of nurses. They employed a two-stage mixed integer programming. In the first stage, the objective function was replaced by another objective that minimised the completion time. This objective value was used as a new scheduling horizon deadline for the second stage that covered the original objective function. According to experiments based on many instances, they found that the increase in the number of patients and caregivers also increased the complexity of the problem while the increase in the number of patients who needed more than one worker decreased the complexity of the problem. Moreover, the mixed integer programming could not solve several large instances with 15 caregivers and 7 visits per caregiver.

[Liu et al., 2017] stated a routing and scheduling model for HHC workers. They considered lunch breaks, qualification levels, and time windows as constraints. Their aim was to minimise total travel times and the number of unscheduled tasks. They proposed a three-index mathematical model which was decomposed into a master problem and several pricing sub-problems, and was optimally solved by a branch-and-price algorithm. It turned out that the branch-and-price algorithm provided superior results in reasonable times.

#### 2.1.1.2 Heuristic/Metaheuristic Solution Methods

Since exact solution methods are limited to very small instances due to their computational complexity, many authors have employed heuristic/metaheuristic methods. Some authors have proposed a singe heuristic/metaheuristic method and compared results with companies' existing results provided by manual routing and scheduling. Moreover, some authors have tested several heuristic/metaheuristic methods and compared results.

[Di Gaspero and Urli, 2014] focused on finding an optimal multi-day HHC schedule by employing a two-stage solution approach. First, they used constraint programming to solve the vehicle routing problem. Next, they introduced a large neighbourhood search method to improve the initial solution provided by constraint programming. This method was applied to solve a set of random instances that mimic a real-world HHC assignment problem. Experimental outcomes showed that the large neighbourhood search significantly improved the constraint programming solution in terms of number of unscheduled patients. However, constraint programming is a better way to reduce the total travelling distance.

[Eveborn et al., 2006] considered a staff planning problem in Sweden where local authorities provided HHC to elder people. They developed a software, LAPS CARE, to help decision makers when they were planning under some soft and hard constraints. Some important components such as databases, maps, optimization routines, and report possibilities were integrated into the system. They proposed a set partitioning model for the problem and a repeated matching algorithm for the solution. Their objective was to find optimal schedules. They showed LAPS CARE's usefulness to save time during visiting and to increase customers' satisfaction. They reported that the software was operational for many local authorities in Sweden due to its user-friendly interface.

[Steeg and Schroder, 2008] minimised the number of different nurses that served each patient. Their first aim was to provide "continuity of care", which basically meant that satisfaction of patients increased if they weren't served by different nurses each time. Additionally, they made a periodic model by considering a one-week schedule horizon. A hybrid approach was employed by combining constraint programming and the large neighbourhood search metaheuristic. Performance of their algorithms with some randomly generated data demonstrated that the large neighbourhood search method worked well.

[Akjiratikarl et al., 2007] examined the scheduling problem of HHC staff by using particle swarm optimization (PSO), which is a collaborative population-based metaheuristic. They targeted to minimise the distance travelled with satisfaction of capacity and service time window constraints. Due to the continuous nature of PSO, it was modified to be suitable for vehicle routing problems with time windows. Therefore, the technique also became appropriate for the discrete assignment problem. The Earliest Start Time Priority with Minimum Distance Assignment, initial solution generator, and local improvement procedures, a method that prevented the algorithm to get stuck in local optima, were developed to increase solution quality. The algorithm produced superior outcomes compared to the existing manual approaches and results by the AiMES Centre at the University of Liverpool employing ILOG.

[Duque et al., 2015] constructed a decision support system for a social profit organisation that provides HHC in Belgium. They modelled the problem as a bi-objective optimisation model considering two different objectives, satisfying preferences of both nurses and patients and minimising total travel distances. Consistency and periodicity of visits, different visits frequencies depending on patient needs, and caregiver absence were taken into consideration in the model. They suggested a two-stage approach based on the first maximising the most crucial objective, the satisfaction of patients and nurses' preferences independent of minimising travel distance. At the second stage, the travel distance was minimised with a constraint on worsening the first objective value below a predefined tolerance limit.

[Hiermann et al., 2015] considered HHC scheduling problem with nurse-patient preferences, time windows, qualifications, and pre-allocated jobs for a home care company operating in Austria. They aimed to minimise the tour length when considering satisfaction of patients and staff. A two-stage approach was employed to solve the problem. At the first stage, an initial solution was created by randomly or constraint programming. At the second stage, the initial solution was iteratively optimised by applying one of four metaheuristics: a memetic algorithm, scatter search, variable neighbourhood search, and Simulating Annealing (SA) hyper-heuristic. Results showed that the memetic algorithm and variable neighbourhood search provided superior results.

[Issaoui et al., 2015] solved HHC problem by considering multiple objectives. They aimed to minimise travel time while maximising patients' satisfaction and the number of visits. A three-phase metaheuristic based on a variable neighbourhood descent and longest processing time algorithms were proposed as solution methodologies. The longest processing time algorithm solved the assignment problem at the first stage. The variable neighbourhood descent algorithm found the shortest path for patients assigned to nurses at second phase. Finally, patients' satisfaction was maximised by using a heuristic that swapped unsatisfied patients between nurses by considering distances calculated in the final stage.

[Redjem and Marcon, 2016] presented a home care service problem. Their problem covered multiple visits to the same patient per day and temporal dependencies between some tasks. They aimed to minimise waiting and travel time of caregivers under hard time window constraints. They presented a two-stage caregiver routing heuristic where the shortest travel time was found for each caregiver without coordination of patients and sequencing restrictions at the first stage while all assumptions and constraints were integrated into the final solution at the second sage. According to their results, their algorithm was very efficient in terms of computational time while it was not sensitive enough to temporal dependencies.

[Rest and Hirsch, 2016] introduced a HHC model which considered daily basis routing and scheduling. Their objective was to minimise travel and waiting times. They considered maximum daily working time with shifts, breaks, clients and workers' satisfaction factor, and caregivers' qualification levels as constraints. Public transportations such as bus, train, and subway were taken into account to model travelling of caregivers during their visits and real time tables derived from public transport service were used when calculating time-dependent travel times. They proposed three Tabu Search (TS) based solution methods for the scheduling problem.

[Lüers and Suhl, 2017] considered the home healthcare problem with a rolling planning horizon. Their aim was to develop a multi-period plan considering continuity of care for patients in terms of visit times and the nurse. At the same time, the plan was flexible to be able to integrate some changes in preferences of both patients and nurses, and in demands or capacity during the execution. They covered skill matching, work regulations, staff satisfaction, and time windows as constraints in their problem when aiming to minimise travel time and penalties regarding to unassigned jobs. They proposed an adaptive large neighbourhood search heuristic as a solution method.

[Hewitt et al., 2016] proposed a HHC problem with longer consistent episodes of care which lasted between two and three months for each patient. They compared three different strategies, weekly basis assignment with perfect information, long term assignment with perfect information, and long term assignment with uncertainty. The first was basically a rolling horizon method that scheduled patients at the beginning of each week and these assignments were passed on to following weeks during episode of care of each patient. New patients were scheduled accordingly. In the second, all visits were scheduled at the beginning of the service horizon. The last strategy considered both known patients and potential future patients derived from historical data when scheduling visits at the beginning of the total service horizon. Future patients were inserted into the schedule as dummy requests and removed when a close actual request arrived. Results showed that long term assignments were superior compared to weekly basis schedules while considering uncertainty was useful for small number of expected patients.

[Shao et al., 2012] constructed weekly schedules for the nurses by minimising overtime and travel cost. A two-phase greedy randomized adaptive search procedure with distinct patient classes was used to solve the problem. In phase I, they found daily routes for nurses and then combined them for weekly schedule while a neighbourhood search algorithm sought the optimal solution in Phase II. Finally, the algorithm proved its superiority on both real data derived from US rehabilitation agency and associated random samples.

[Trautsamwieser and Hirsch, 2011] presented a Variable Neighbourhood Search (VNS) solution method for a daily planning of HHC services. Their objective was to minimise nurse travel times and dissatisfaction levels of both patients and caregivers. They included some constraints such as appropriate assignments of nurses to patients based on skill level, language match, declinations, daily and weekly working times, hard time windows, and breaks. The proposed solution methodology was tested with generated data and a real life data set provided by Austrian Red Cross. According to results, travel times could be decreased by up to 45% with the proposed solution method.

[Bertels and Fahle, 2006] developed a software optimizing both rostering and routing problem simultaneously while considering different hard and soft constraints. The software employed a combination of linear programming, constraint programming, and metaheuristics to maximise staff and patients' satisfaction and minimise transportation cost. They found hybrid approaches such as a combination of TS and constraint programming were superior to single paradigms such as TS or SA.

[Lin et al., 2017] presented a HHC model with two problems. In the first problem, they considered rostering and routing of nurses by satisfying time windows, qualification levels, preferences of nurses, and work regulations such as breaks, maximum work hours, and holidays. In the second problem, re-rostering of the current schedule was considered due to visit time changes of patients and absences of nurses or patients by minimising the difference between the original and new schedules. They proposed a modified harmony search algorithm. The experimental and statistical analysis showed that the modified harmony search provided good results in shorter times compared to the standard harmony search algorithm.

#### 2.1.1.3 Comparative Solutions

Although heuristic/metaheuristic methods provide solutions for large instances in reasonable times, they can never guarantee optimal results. Comparison of results of these algorithms with each other or with results of manual plans does not show their performance properly. Therefore, many authors have developed both exact and heuristic/metaheuristic solution methods to be able to show performance of heuristic/metaheuristic algorithms by comparing them with exact methods under same problem settings with smaller instances.

[Rasmussen et al., 2012] presented a home care crew scheduling problem with soft patient's nurse preference restrictions and temporal dependencies as synchronisation, two nurses needed to visit a patient at the same time, and minimum-maximum difference, a nurse started a duty after another nurse finalized it. The problem was modelled as a set partitioning problem by adding temporal dependencies as generalised precedence constraints. They used a branch-and-price algorithm and a novel visit clustering approach based on the soft preference restrictions. The application of the algorithm to a real-life problem and examples derived from realistic settings showed that the visit clustering approach provided solutions for larger problems when the branch-and-price algorithm could not find optimal solutions.

[Bard et al., 2014] constructed weekly schedules of HHC staff servicing in 135 nursing homes. They tried to minimise cost over a 5-day planning horizon under over time rules, breaks, and time window constraints. Additionally, preferences of patients and nurses were taken into account unless they violated feasibility of the model. They modelled the problem as a large-scale mixed integer program and used a branch-and-price-and-cut algorithm to solve it. Furthermore, a rolling horizon algorithm was used to find solutions for larger instances since the branch-and-price-and-cut algorithm was slow to converge. They employed data and regulations such as the practices, policies, legal restrictions, and compensation rules of Key Rehab, a company providing physical, occupational, and speech therapy in US Midwest.

[Braekers et al., 2016] proposed a bi-objective optimisation model to examine the trade-off between operating cost covering overtime and travel costs and service level including preferences of clients and nurses. They solved the problem with a metaheuristic algorithm based on a multi-directional local search framework. They conducted computational experiments by using several benchmark problem samples generated based on a real data set. The algorithm performed quite well compared to exact solution methods for small size instances. The results showed that allowing for an additional operating cost was able to improve the service level significantly.

[Liu et al., 2013] proposed a HHC problem based on pick up and delivery from depots and hospitals to patients or vice versa. According to their model, vehicles delivered drugs and medical devices from the HHC company to patients' homes, delivered special drugs from a hospital to patients, pick up of bio samples and unused drugs and medical devices from patients to deliver the hospital again in assigned time windows. They proposed two mixed integer programming models, a Genetic Algorithm (GA), and a TS method. Exact methods failed to find optimum solutions in the given time interval while metaheuristic methods provided solutions in reasonable times based on different test instances in the literature.

[Zhan et al., 2015] studied an HHC routing and appointment scheduling problem with uncertain service times for a doctor. Their objectives are to minimise patients' waiting times, the doctor's idle time, and total travel time. First, they solved a small size problem with a mixed integer programming under the assumption of known patients' service time distributions. Next, the problem was modelled as a two-stage stochastic programming problem and the L-Shape method was used since the branch-and-cut algorithm was not able to solve the problem in a reasonable time for larger instances. Finally, they suggested a heuristic method which could calculate approximate cost of idle and waiting times just by considering the predecessor's random service time. Results showed that the heuristic provided good results for large size problems.

[Yuan et al., 2015] suggested a stochastic programming model for HHC scheduling and routing problems with stochastic service times. They aimed to minimise caregiver and service costs and late arrival penalty by considering multiple nurses with different skill levels. Patients were categorised into their medical needs and could be served only by a nurse who has an adequate skill level for the treatments. A column generation method and label algorithm were proposed to solve master and pricing sub-problems.

[Trautsamwieser and Hirsch, 2014] developed a solution method for a medium term HHC planning problem in which the planning horizon lasts a week. Only adequately skilled nurses could serve patients who needed special treatments one or multiple times in the week. Visits had to be done in the given time windows and same times in the week if patients required several visits. Nurses were required to have a break after working a certain number of hours and not to work longer than a given weekly number of hours. Their objective was to minimise waiting and travelling times. They introduced a Branch-Price-and-Cut algorithm as a solution methodology. Furthermore, VNS metaheuristic was proposed to solve the problem in short computational times.

[Riazi et al., 2014] presented a mixed integer linear programming for HHC routing and scheduling problem. The model includes some constraints such as pre-specified time windows for visits and nurse skill levels. Their objective was to minimise total travel times. They implemented several solution methodologies, the centralized method, the logic-based Benders decomposition, the gossip algorithm, and its extensions. The logic-based Benders decomposition method decomposed the problem into task assignment problem (master) and several vehicle routing problems (sub-problems). The gossip algorithm decreased the problem size by dividing the global problem into local problems and solved them independently. Local problems in the model covered a small number of nurses and customers.

[Mankowska et al., 2014] developed a model for daily HHC routing and scheduling. The model covered nurse qualifications, patients' preferences, interdependent and dependent services where the former requires that some tasks must be handled before other tasks and the latter is taking into consideration when a task needs more than one worker. They aimed to minimise travel and idle times of nurses and provide fair allocation of waiting times among the requests. They introduced a mixed integer linear model and solved a small size problem with ILOG Cplex Solver and a large size instance with an adaptive VNS algorithm.

[Guericke and Suhl, 2017] developed a HHC model mainly considering work regulations and legal requirements. They took into account break times, weekly work durations, and shift rotations according to laws and regulations in Germany to be able to investigate their influence on results. They proposed a mixed integer linear solution for a small size problem setting. Moreover, an adaptive large neighbourhood search based heuristic was provided to cope with real-size complex problems in a reasonable computational time. According to results, the heuristic method showed a good performance compared to the mixed integer program in a relatively short time.

[Frifita et al., 2017] developed a model for a HHC problem with time windows and synchronization which meant multiple caregivers visit a patient at the same time. They proposed a general VNS method to be able to minimise travel times of caregivers. The proposed methodology was compared to a mixed integer model and a heuristic method for a variety of real life instances. According to results, their method was fast compared to the mixed integer model and provided results close to the optimal solution.

[Du et al., 2017] presented a HHC scheduling optimization problem with known demands and service capabilities. Their aim was to minimise the total service cost that included travel, service, and penalty costs while considering qualification matches, time windows, and service priority based on seriousness of the patients' conditions. They developed an integer programming model and proposed a GA with local search method in order to solve the problem. They compared results of the proposed solution method with a commercial software for a case study in China. It turned out that the GA with local search method provides fast and reasonable results for the real life data.

[Triki et al., 2014] introduced a periodic HHC problem consisting weekly and daily routing and scheduling plans. They proposed a two-phase method that optimised routes and schedules in a week according to known weekly demand in the first stage and optimised daily routes by minimising deviations from the weekly route in order to assign new requests in the second stage. Qualification levels, time windows, and lunch breaks were considered as constraints. They presented a mixed integer programming for small size instances and a TS method for large size instances in order to minimise the total routing cost and the exceeding workload. Results showed that the TS method ensured good solution quality for large size instances for which CPLEX failed to find any feasible solution.

[Arabzadeh et al., 2016] stated a weekly HHC planning problem. They aimed to minimise travel times of caregivers and delays in visits of patients. Time windows, qualifications of nurses, interval times between two consecutive visits of patients on the same day, and working times of part time and full time workers were considered as constraints in the problem. They proposed mixed integer programming for small scale data and a GA and an "Imperialist Competitive Algorithm" for large size problems. The metaheuristic methods provided near optimal solutions for small instances while finding solutions in reasonable times for large instances for which GAMS failed to find any feasible solution.

[Decerle et al., 2016] presented a daily routing and scheduling problem in HHC. Their aim was to minimise travel time and work time costs. Main contribution of their study was to consider nurses with higher salaries and unlicensed assistant workers with lower salaries separately in order to generate more cost effective schedules for shared visits that need a couple of workers. They proposed a two-phase metaheuristic method. In the first phase, nurses were optimally scheduled while unlicensed assistant workers were assigned to shared and the remaining visits in the second phase. They also developed a mixed integer programming model to provide global solutions by considering all staff simultaneously. Results showed that the metaheuristic ensured fast and near optimal solutions compared to the exact method.

[Tozlu et al., 2015] presented a HHC routing problem with crew constraints. In the problem, there were two types of workers, nurses and aides, who visited patients together or separately. Three different objective functions, minimising travel times, the total number of staff, and the total number of vehicles, were defined in the model. They proposed a mixed integer programming model and VNS for the solution. According to results based on different size instances, the VNS was able to find quite good and fast results compared to the results given by CPLEX.

#### 2.1.1.4 Why is a Heuristic Method Preferred for Our Problem?

As mentioned before, exact solution methods can provide optimal solutions and work for small instances in reasonable computational times. The majority of studies have employed only heuristic/metaheuristic methods to be able to cope with real data or both exact and heuristic/metaheuristic methods to be able to show how heuristic/metaheuristic methods perform compared to exact methods under same problem settings. SBA is also a heuristic based method which cannot guarantee optimal solutions. Therefore, one can ask why we employ a heuristic method instead of an exact method and why we even do not use one of exact methods for a very small instance to make comparison with results of SBA and show our algorithm's performance. Existing papers as we mentioned above, generally focused on static problem settings for which the number of patients, their locations, the number of weekly visits were already known. Even under this deterministic setting, exact methods work only for small instances. In our problem, patients arrive dynamically and details about the patient locations and their needs are only revealed over time. This also makes the problem more complicated. First, when we consider tens of nurses, hundreds of patients, and thousands of visits in a-year simulation horizon, it is quite obvious that exact solution methods such as multi-stage stochastic programming or stochastic dynamic programming would fail to find even a solution. Moreover, one of expectations of this study is to help decision makers to respond requests of patients as soon as they arrive. Therefore, it is unacceptable that patients wait responses for long hours when we employ exact methods. Next, decisions in dynamic problems are made based on optimal expected outcomes and solutions depend on generated scenarios. Therefore, modelling our problem with one of the exact solution methods does not give an optimal solution that can show upper or lower bounds for benchmark purposes. Lastly, as we will explain in detail later, the only study that is very close our perspective is study of [Bennett and Erera, 2011]. We already mimicked their algorithms and compared their results with ours.

### 2.1.2 Dynamic Problems

As we mentioned above, existing studies in the literature generally focused on static problem settings for which the number of patients was already known, but requests arrive to the system dynamically during service horizon in practice. Additionally, they did not consider any acceptance policy. We have found only the studies of [Bennett and Erera, 2011] and [López-Santana et al., 2016] which consider dynamic patient sets. [López-Santana et al., 2016] proposed a HHC caregivers daily scheduling problem. The problem is dynamic since a patient assignment decision had to be made as soon as the patient request arrived. They employed an agent based simulation method to model attributes of patients and caregivers, and mixed integer programming model to find caregivers optimal routing schemes. Their aim was to minimise the total travel time and the service promise factor depending on visiting patients in specific time windows defined based on their priority levels. They considered qualification level and working time constraints in the model. Several scenarios based on the number of caregivers, time periods, and coefficients in the objective function were tested. Although their model is dynamic, the concept of their problem is quite different from ours. Their problem is very similar to pick up and delivery problems where new customer requests arrive when the current plan is being executed and they should be scheduled in the day if possible. However, we start to schedule visits of an accepted patient in the next week and service horizon of each patient is at least 4 weeks. Furthermore, they employ a simulation method and mathematical model which restrict their study with several scenarios and limited number of nurses, patients, and simulation time. [Bennett and Erera, 2011] presented a myopic planning approach for the single nurse HHC problem. This approach proposed a capacity based insertion heuristic when integrating a new patient request to the existing schedule by considering the nurse's remaining available time explicitly. Furthermore, they modelled the problem as dynamic periodic fixed appointment time, which means that patients arrived dynamically and they were assigned to predetermined days over a predetermined number of weeks to visit. Their objective was to maximise the number of patients being served by a nurse. However, the proposed distance and capacity based heuristics are greedy algorithms which try to choose the best movement whenever a new request arrives without considering future requests or only partially considering. Moreover, these heuristics accept all requests and ignore that to reject a request now can allow to accept more requests in the future. The point behind an acceptance or a rejection decision is that if a request of a patient located far from the tour is rejected, more closer requests in the future can be assigned to the tour. In other words, we spend time serving patients instead of travelling between distant locations. Of course, we should project future demand properly to make this decision. Therefore, we tried to answer two questions in this study. First, do we accept or reject the request? And second, if we decide to accept the request, which visit days and time slots should it be assigned to?

## 2.2 Dynamic Vehicle Routing Problem (DVRP) and Periodic Dynamic Vehicle Routing Problem (DPVRP) Studies

In contrast to the classical VRP, real-world applications often force decision makers to design routing plans online where the visit of next customer is decided as soon as it becomes available. This is where DVRP is taken into consideration. DVRP studies begin with [Wilson and Colvin, 1977]. They employed a greedy insertion heuristic to put dynamically arriving requests into a tour for a single vehicle. Readers can find detailed literatures reviews on DVRP in [Thomas, 2010], [Ritzinger et al., 2016], and [Pillac et al., 2013]. Because DVRP literature is vast, we only discuss some papers whose solution methods are related to our solution methodology. [Yang et al., 2000] considered restocking by returning to the depot when a stockout occurs or in anticipation of a stockout. They developed two heuristics for single vehicle and multiple vehicles to minimise total travel cost. [Secomandi, 2000] compared the performance of two neuro-dynamic programming algorithms, optimistic approximate policy iteration and a roll-out policy for DVRP. According to their results, the roll-out policy performed better for vehicle routing applications when dynamism is taken into consideration. [Larsen et al., 2002] described the degree of dynamism concept to select a suitable algorithm and models depending on the dynamic features of the system and explored its effectiveness for DVRP and similar problems. They applied different degree of dynamism to a Partially Dynamic Travelling Repairman Problem. Results showed that an increasing degree of dynamism caused a linear increase in tour lengths. [Ichoua et al., 2006] suggested a TS based solution method to exploit probabilistic knowledge about future request arrivals. They proposed a waiting strategy where vehicles wait at their current locations based on knowledge about future requests if there is a time gap until the next customer service. [Hvattum et al., 2006] proposed a multi-stage stochastic programming model and a heuristic solution methodology. The heuristic generated scenarios including scheduled visits and random customers raised from known distributions. Each sample scenario was solved as a deterministic VRP and common features in the sample scenario solutions were employed to construct routes. [Bent and Van Hentenryck, 2004] modelled DVRP with time windows and aimed to maximise the number of daily visits. They proposed a multiple scenario approach based on generating routing plans including both known and future customers. A distinguished plan selected by a consensus function in terms of the smallest travel cost was employed for decision making processes. The multiple scenario approach was tested against greedy approaches under dynamism varying between 30% and 80%. The main difference between the solution methods of Bent et al. and Hvattum et al. is that the multiple scenario approach from [Bent and Van Hentenryck, 2004] works as TS with adaptive memory by maintaining and updating routing and distinguished plans consisted of current and future customers while the heuristic of [Hvattum et al., 2006] is a multi-stage model in which each stage represents a time interval over the time horizon. The aim is to find a plan that minimises the expected cost of visiting both current and future requests at the beginning of each stage.

Although the problem we consider is certainly related to the dynamic vehicle routing problem, there are also substantial differences. The typical paper on dynamic VRP considers a single day, and customer requests arriving while vehicles are already under way. The customer requests then have to be integrated into the existing tours, but tours can usually be changed dynamically. On the other hand, in our problem we assume all customer requests arrive in the week before the first service, they arrive dynamically, and we have to commit to fixed appointment dates and times for each request when it arrives. Also, while usually DVRP problems assume a customer request only has to be serviced once, we assume patients have to be serviced several times a week, over several weeks, and at the same times and days every week.

It is important to refer online problems and algorithms when discussing DVRP. An online algorithm is simply a method generating solutions at any state of a problem without knowing the entire input. In this sense, it looks like DVRP but there are some differences. In DVRP, the distribution of interarrival times and the distribution of locations of patients or customers and historical data are partially or fully known a priori. Furthermore, the proportion of known customers to immediate customers, called “degree of dynamism”, and arrival times that directly affects the “effective degree of dynamism”are important factors to increase or decrease the complexity of DVRP [Larsen et al., 2008]. However, online algorithms are more conservative approaches and are usually applied to real-time environments where there are no known distributions and advanced information [Jaillet and Wagner, 2008]. A new taxi or delivery company in a city can be an example [Bertsimas et al., 2018]. Online algorithms are assessed by competitive analyses. Competitive ratios are defined as the performance of an online algorithms to the performance of an optimal offline algorithms in where all necessary data are known before optimisations start [Albers, 2003]. Although the competitive analysis framework could be used for evaluation of algorithms in DVRP, it works only for simple problem sets since real life constraints such as time windows make problems very complex for competitive analyses [Larsen et al., 2008]. Moreover, in DVRP, it is hard to provide a competitive ratio when considering many scenarios caused by different distributions during the optimisation. On the contrary, a competitive ratio developed for a problem such as online TSP can be compared to other online algorithms developed by other researchers. Overall, our current problem setting is closer to the DVRP concept since we know distributions of interarrival times, locations, and expected number of weekly visits in our problem.

DPVRP considers several visits requested by one customer in the planning horizon. The main challenge is whether or not to postpone a visit of a customer to another allowable day. The aim of decision is to construct shortest tours today as well as consecutive days with potential future and postponed requests. The literature on the DPVRP is very scarce. [Wen et al., 2010] defined a model that a given number of vehicles serviced orders that were accumulated during a day and had to be serviced by starting a day after. The problem was to determine orders that could be serviced as soon as possible or should be delayed next consecutive days to be able to construct tours which minimised travel cost and customer waiting time. There was no time window for daily visits. Routes for each day in the planning horizon were constructed based on the orders known so far and the routes were fixed before their execution. [Angelelli et al., 2007] similarly outlined a problem that customers had to be visited in next two consecutive days after their orders were taken. The problem was to decide which customers had to be served and whose service could be postponed. The aim was to minimise total travelled distance during the service horizon. They tested several simple algorithms in terms of the length of planning horizon and the location of customers. [Albareda-Sambola et al., 2014] examined routing and delaying decisions for the problem where locations of future customer requests were known probabilistically. Although the problem frame was similar to studies of [Wen et al., 2010] and [Angelelli et al., 2007], [Albareda-Sambola et al., 2014] used probabilistic data to reduce cost and improve solution quality. The decision of service in the current time period or delay for the next time period was made based on the profit of the visit determined by urgency of the service and convenience of waiting to be able to visit future requests. They compared their results with two simple strategies based on visiting customers at the beginning or end of their service windows.

In our problem, we consider multiple visits belonging to the same patients and assume that patient requests arrive dynamically. In this sense, our problem seems similar to DPVRP. However, exact service times and visits days are decided as soon as patient requests arrive and they cannot be changed according to the condition of tours. DPVRP studies we mentioned above are only restricted in terms of the time interval that visits of a customer have to be performed. On the other hand, they solve the routing problem with fully known data set for each day and only decide on which visits are performed for that day or postponed to another allowed days. [Wen et al., 2010] and [Angelelli et al., 2007] do not employ any prediction method for future customer requests while [Albareda-Sambola et al., 2014] employ historical data sets to improve solution quality. However, we develop a strategy that generates scenarios mimicking future patient requests in order to find the most suitable visit days and times for a patient.

## 2.3 Other Studies

[Hertz and Lahrichi, 2009] aimed to balance the workload of nurses and additionally to minimise long travels. They analysed and used the data of Health and Social

Services Centres in Montreal, Canada. The problem was modelled by a mixed integer programme with some non-linear constraints and objectives. TS algorithm was employed to the solve problem. Moreover, they compared results of TS algorithm to results of CPLEX solver after removing some non-linear constraints to indicate goodness of TS. Results showed that the TS algorithm solved the problem effectively. They concluded that providing similar workload and avoiding overload for some nurses highly depended on a careful partitioning of the territory.

[Milburn et al., 2012] examined the indirect supply chain cost on HHC by conducting a questionnaire for home health care agencies in US. According to the data analysis, they found indirect cost, which included ordering, storing, handling, delivering supplies, could become high under high patient volume and agency affiliation factors. Therefore, they advised that nurse involvement in non-clinical duties such as ordering, sorting, and picking supplies should be reduced as much as possible.

[Chahed et al., 2009] presented an anti-cancer drugs supply chain problem, for which the anti-cancer drug had to be prepared in health centers because of a recent French health regulation and they had to be delivered to patients under specific conditions and considering drugs' shelf life. They tried to minimise the total travel times under consideration of production starting time and time windows for visits. CPLEX was employed to solve the standard integer model under limitation of ten patients per day.

[Marcon et al., 2017] proposed a solution methodology based on simulating caregiver behaviour by using Agent Based Simulation. They developed four decision rules for caregivers that were used right before visiting a patient. Nearest Patient Rule was autonomous caregiver behaviour where caregivers could choose the next patient by themselves without following any routes. Next, Shortest Route Rule was that caregivers had to follow a planned route. No-wait Route Rule was that caregivers had to follow the planned route unless the next patient was unavailable. Finally, Balanced Route Rule was similar to the previous rule but the fact that caregivers could return the first patient at the list if the second was not available

Table 2.1: A classification of publications in terms of performance measures and objectives

| | Travel Time/Cost | Waiting Time/Cost | Patient/Staff Preferences | Unscheduled Patient/Task |
|---|---|---|---|---|
| [Begur et al., 1997] | ✓ | | | |
| [Di Gaspero and Urli, 2014] | ✓ | ✓ | | ✓ |
| [Bard et al., 2014] | ✓ | ✓ | | |
| [Carello and Lanzarone, 2014] | ✓ | | | |
| [Cappanera and Scutellà, 2014] | ✓ | | | |
| [Duque et al., 2015] | ✓ | | ✓ | |
| [Zhan et al., 2015] | ✓ | ✓ | | |
| [Liu et al., 2013] | ✓ | | | |
| [Hiermann et al., 2015] | ✓ | | ✓ | |
| [Braekers et al., 2016] | ✓ | ✓ | ✓ | |
| [Bennett and Erera, 2011] | | | | ✓ |
| [Mankowska et al., 2014] | ✓ | ✓ | | ✓ |
| [Issaoui et al., 2015] | ✓ | | ✓ | |
| [Redjem and Marcon, 2016] | | ✓ | | |
| [Guericke and Suhl, 2017] | | ✓ | | |
| [Wirnitzer et al., 2016] | | | ✓ | ✓ |
| Our study | | | | ✓ |

| | Travel Time/Cost | Waiting Time/Cost | Patient/Staff Preferences | Unscheduled Patient/Task |
|---|---|---|---|---|
| [Lin et al., 2017] | ✓ | | ✓ | |
| [Lüers and Suhl, 2017] | ✓ | | | ✓ |
| [Masmoudi and Mellouli, 2014] | ✓ | ✓ | | |
| [Decerle et al., 2016] | ✓ | | | |
| [Nasir and Dang, 2017] | ✓ | | | |
| [Tozlu et al., 2015] | ✓ | | | |
| [Liu et al., 2017] | ✓ | | | ✓ |
| [Hewitt et al., 2016] | ✓ | | | |
| [López-Santana et al., 2016] | ✓ | ✓ | | |
| [Yalçındağ et al., 2016a] | ✓ | | | |
| [Issabakhsh et al., 2018] | ✓ | | | |
| [Frifita et al., 2017] | ✓ | | | |
| [Du et al., 2017] | ✓ | ✓ | | |
| [Triki et al., 2014] | ✓ | | | |
| [Arabzadeh et al., 2016] | ✓ | | ✓ | |
| [Rest and Hirsch, 2016] | ✓ | ✓ | | |
| [Manerba and Mansini, 2016] | | | | ✓ |
| Our study | | | | ✓ |

| | Travel Time/Cost | Waiting Time/Cost | Patient/Staff Preferences | Unscheduled Patient/Task |
|---|---|---|---|---|
| [Steeg and Schroder, 2008] | ✓ | ✓ | ✓ | |
| [Bertels and Fahle, 2006] | ✓ | | ✓ | |
| [Eveborn et al., 2006] | ✓ | | ✓ | |
| [Akjiratikarl et al., 2007] | ✓ | | | |
| [Rasmussen et al., 2012] | ✓ | | ✓ | ✓ |
| [Shao et al., 2012] | ✓ | | | |
| [Trautsamwieser and Hirsch, 2014] | ✓ | ✓ | | |
| [Riazi et al., 2014] | ✓ | | | |
| [En-nahli et al., 2015] | ✓ | ✓ | | |
| [Trautsamwieser and Hirsch, 2011] | ✓ | | ✓ | |
| Our study | | | | ✓ |

as well. They evaluated performances of decision rules according to five criteria: efficiency, pertinence, scalability, robustness, and implementability. According to results, Balanced Route Rule outperformed in terms of minimising travel times.

[Nasir and Dang, 2017] presented a HHC resource dimensioning and assignment problem consisted of determining patient group based clustering, the number of HHC offices and workers, routing and scheduling of patients simultaneously. Their objective was to minimise the total travel times between patients and workers as well as offices and workers. They proposed a mixed integer programming model and tested four different scenarios based on relaxation of some cost factors in the objective function and constraints. It turned out that the model worked well with small size instances.

[Nguyen et al., 2015] addressed an uncertainty problem on availability of the nurses in HHC. They minimised costs raised from travel and waiting times, and hiring external caregivers due to unavailability of caregivers in the company. They used time windows and different skill levels as constraints. A robust optimisation approach by taking different conservativeness degrees into account was proposed. They used a metaheuristic solution method based on a GA and mathematical programming.

[Shi et al., 2017] presented a HHC problem with fuzzy demand and time constraints. Their aim was to minimise the total driving times of vehicles that delivered medical drugs patients needed. The main challenge was to carry enough drugs whose quantity were uncertain when constructing tours in order to prevent returning vehicles to the depot. A fuzzy chance constraint was defined based on the fuzzy credibility theory. They proposed a hybrid GA integrated with stochastic simulation methods as a solution. They compared results of the proposed model with a mixed integer mathematical model and it turned out that the proposed model worked well for small and large instances.

[Rodriguez et al., 2015] addressed a problem of staff dimensioning in HHC. They minimised the number of HHC workers with different skills under uncertain demand

Table 2.2: A classification of publications in terms of constraints

| | Skill Matching | Multi Worker | Time Windows | Consistency/Periodicity | Patient/Staff Preferences | Breaks |
|---|---|---|---|---|---|---|
| [Begur et al., 1997] | ✓ | | ✓ | ✓ | | |
| [Di Gaspero and Urli, 2014] | | | ✓ | | | |
| [Bard et al., 2014] | | | ✓ | | ✓ | ✓ |
| [Carello and Lanzarone, 2014] | | | ✓ | ✓ | | |
| [Cappanera and Scutellà, 2014] | ✓ | | ✓ | ✓ | | |
| [Duque et al., 2015] | ✓ | | | ✓ | | ✓ |
| [Zhan et al., 2015] | | | ✓ | | | |
| [Liu et al., 2013] | | | ✓ | | | |
| [Hiermann et al., 2015] | ✓ | | ✓ | | ✓ | |
| [Braekers et al., 2016] | | | ✓ | | | |
| [Bennett and Erera, 2011] | | | | ✓ | | |
| [Mankowska et al., 2014] | ✓ | ✓ | ✓ | | | |
| [Issaoui et al., 2015] | | | ✓ | | | |
| [Redjem and Marcon, 2016] | | ✓ | ✓ | | | |
| [Rest and Hirsch, 2016] | ✓ | | ✓ | | ✓ | ✓ |
| [Manerba and Mansini, 2016] | ✓ | | ✓ | | | |
| Our study | ✓ | | | ✓ | | |

| | Skill Matching | Multi Worker | Time Windows | Consistency/Periodicity | Patient/Staff Preferences | Breaks |
|---|---|---|---|---|---|---|
| [Masmoudi and Mellouli, 2014] | | ✓ | ✓ | | | |
| [Decerle et al., 2016] | ✓ | ✓ | ✓ | | | |
| [Nasir and Dang, 2017] | ✓ | | ✓ | | | |
| [Tozlu et al., 2015] | ✓ | ✓ | ✓ | | | |
| [Liu et al., 2017] | ✓ | | ✓ | | | ✓ |
| [Hewitt et al., 2016] | | | ✓ | ✓ | | |
| [López-Santana et al., 2016] | ✓ | | ✓ | | | |
| [Guericke and Suhl, 2017] | ✓ | | ✓ | | | ✓ |
| [Wirnitzer et al., 2016] | ✓ | | ✓ | ✓ | | ✓ |
| [Yalçındağ et al., 2016a] | ✓ | | ✓ | ✓ | | |
| [Frifita et al., 2017] | ✓ | ✓ | ✓ | | | |
| [Du et al., 2017] | ✓ | | ✓ | | | |
| [Triki et al., 2014] | ✓ | | ✓ | | | ✓ |
| [Arabzadeh et al., 2016] | ✓ | | ✓ | ✓ | | ✓ |
| [Lin et al., 2017] | ✓ | | ✓ | ✓ | | ✓ |
| [Lüers and Suhl, 2017] | ✓ | | ✓ | ✓ | ✓ | ✓ |
| Our study | ✓ | | | ✓ | | |

| | Skill Matching | Multi Worker | Time Windows | Consistency/Periodicity | Patient/Staff Preferences | Breaks |
|---|---|---|---|---|---|---|
| [Steeg and Schroder, 2008] | | | ✓ | ✓ | | |
| [Bertels and Fahle, 2006] | ✓ | | ✓ | | ✓ | |
| [Eveborn et al., 2006] | ✓ | | ✓ | ✓ | ✓ | ✓ |
| [Akjiratikarl et al., 2007] | | | ✓ | | | |
| [Rasmussen et al., 2012] | | ✓ | ✓ | | | |
| [Shao et al., 2012] | ✓ | | ✓ | | | |
| [Trautsamwieser and Hirsch, 2014] | ✓ | | ✓ | ✓ | ✓ | |
| [Riazi et al., 2014] | ✓ | | ✓ | | | |
| [En-nahli et al., 2015] | ✓ | | ✓ | | ✓ | ✓ |
| [Trautsamwieser and Hirsch, 2011] | ✓ | | ✓ | | ✓ | ✓ |
| Our study | ✓ | | | ✓ | | |

to be able to serve as many patients as possible with a given performance level. The uncertain demand covered the number of patients, regions where patients came from, and task and durations that patients needed. They presented a two-stage integer linear programme, where minimal resource needs had to be found for each demand scenario at the first phase while the optimal number of employees was calculated to satisfy the performance level.

[Yalçındağ et al., 2016b] presented a HHC patient assignment problem by estimating travel time of staff with kernel regression technique. Their objectives were to balance workloads of caregivers and minimise total travel times. Kernel method predicted travel times by using historical data. The main point behind their study was that making assignment decisions based on minimising only Euclidean distances would not be correct since patient attributes such as availability of a family member, time limitation of treatments, etc. directly affected assignment decisions as well. The data-driven approach based on kernel regression employed workers' specific past patterns to be able to predict travel times. Numerical results showed that their approach was superior to the average value and k-nearest neighbour search methods.

Table 2.3: A classification of publications in terms of solution methodologies

| | Exact | Heuristics | Single Objective | Multi Objective | Static | Dynamic |
|---|---|---|---|---|---|---|
| [Begur et al., 1997] | | ✓ | ✓ | | ✓ | |
| [Di Gaspero and Urli, 2014] | | ✓ | ✓ | | ✓ | |
| [Bard et al., 2014] | ✓ | ✓ | ✓ | | ✓ | |
| [Carello and Lanzarone, 2014] | ✓ | | ✓ | | ✓ | |
| [Cappanera and Scutellà, 2014] | ✓ | | ✓ | | ✓ | |
| [Duque et al., 2015] | | ✓ | | ✓ | ✓ | |
| [Zhan et al., 2015] | ✓ | ✓ | ✓ | | ✓ | |
| [Liu et al., 2013] | ✓ | ✓ | ✓ | | ✓ | |
| [Hiermann et al., 2015] | | ✓ | ✓ | | ✓ | |
| [Braekers et al., 2016] | ✓ | ✓ | | ✓ | ✓ | |
| [Bennett and Erera, 2011] | | ✓ | ✓ | | | ✓ |
| [Mankowska et al., 2014] | ✓ | ✓ | ✓ | | ✓ | |
| [Issaoui et al., 2015] | | ✓ | | ✓ | ✓ | |
| [Redjem and Marcon, 2016] | | ✓ | ✓ | | ✓ | |
| [Rest and Hirsch, 2016] | | ✓ | ✓ | | ✓ | |
| [Manerba and Mansini, 2016] | ✓ | | ✓ | | ✓ | |
| Our study | | ✓ | ✓ | | | ✓ |

| | Exact | Heuristics | Single Objective | Multi Objective | Static | Dynamic |
|---|---|---|---|---|---|---|
| [Lüers and Suhl, 2017] | | ✓ | ✓ | | ✓ | |
| [Masmoudi and Mellouli, 2014] | ✓ | | ✓ | | ✓ | |
| [Decerle et al., 2016] | ✓ | ✓ | ✓ | | ✓ | |
| [Nasir and Dang, 2017] | ✓ | | ✓ | | ✓ | |
| [Tozlu et al., 2015] | ✓ | ✓ | ✓ | | ✓ | |
| [Liu et al., 2017] | ✓ | | ✓ | | ✓ | |
| [Hewitt et al., 2016] | | ✓ | ✓ | | ✓ | |
| [López-Santana et al., 2016] | ✓ | | ✓ | | | ✓ |
| [Guericke and Suhl, 2017] | ✓ | ✓ | ✓ | | ✓ | |
| [Wirnitzer et al., 2016] | ✓ | | ✓ | | ✓ | |
| [Yalçındağ et al., 2016a] | ✓ | | ✓ | | ✓ | |
| [Issabakhsh et al., 2018] | ✓ | | ✓ | | ✓ | |
| [Frifita et al., 2017] | ✓ | ✓ | ✓ | | ✓ | |
| [Du et al., 2017] | ✓ | ✓ | ✓ | | ✓ | |
| [Triki et al., 2014] | ✓ | ✓ | ✓ | | ✓ | |
| [Arabzadeh et al., 2016] | ✓ | ✓ | ✓ | | ✓ | |
| [Lin et al., 2017] | | ✓ | ✓ | | ✓ | |
| Our study | | ✓ | ✓ | | | ✓ |

| | Exact | Heuristics | Single Objective | Multi Objective | Static | Dynamic |
|---|---|---|---|---|---|---|
| [Steeg and Schroder, 2008] | | ✓ | ✓ | | ✓ | |
| [Bertels and Fahle, 2006] | | ✓ | ✓ | | ✓ | |
| [Eveborn et al., 2006] | | ✓ | ✓ | | ✓ | |
| [Akjiratikarl et al., 2007] | | ✓ | ✓ | | ✓ | |
| [Rasmussen et al., 2012] | ✓ | ✓ | ✓ | | ✓ | |
| [Shao et al., 2012] | | ✓ | ✓ | | ✓ | |
| [Trautsamwieser and Hirsch, 2014] | ✓ | ✓ | ✓ | | ✓ | |
| [Riazi et al., 2014] | ✓ | ✓ | ✓ | | ✓ | |
| [En-nahli et al., 2015] | ✓ | | | ✓ | ✓ | |
| [Trautsamwieser and Hirsch, 2011] | | ✓ | ✓ | | ✓ | |
| Our study | | ✓ | ✓ | | | ✓ |

# Chapter 3

# HHC Model for A Singe Nurse

In this Chapter, we develop SBA for a single nurse by proposing two different versions, Daily and Weekly SBA, the former depending on constructing tours for each day of the week independently and the latter considering all visits of requests in the week simultaneously when constructing tours for each day. We empirically compare with two greedy heuristics from the literature, Distance and Capacity Heuristics. Next, we examine how different service time durations and violation of service continuity affect results. Finally, we demonstrate a new pricing policy based on patient preferred visit days and times at the end of this chapter.

All algorithms are coded with Java programming language. We present codes of the Daily Scenario Based Approach in the Appendix as an example.

## 3.1 Problem Statement

The problem we consider is a single nurse HHC scheduling problem in a dynamic environment over a planning horizon.

**Nurse:** All patients are visited by a single nurse in a defined geographic service area. Each working day is divided into equally-spaced time intervals to schedule patient visits. A set of possible appointment times, $K$, can be defined as:

$$K=\{b+i\phi \;:\; i=0,1,\ldots,k\},$$

where $b$ is the earliest time for an appointment and $\phi$ is the time between appointment times. Travel time between patient $i$ and $j$ is denoted by $m(g_i,g_j)$ in minutes where $g_i$ represents the location of patient $i$. All travel times are always rounded up to the nearest multiple of time slot.

**Patients:** Interarrival times between patients' requests are exponentially distributed with known parameters over the planning horizon. A request $i$ from location $g_i$ contains weekly service frequency $f_i$, episode of care $ec_i$ that represents how many weeks patient $i$ needs care, service duration for each visit $sd_i$, starting time for the service $K_i$, and weekly allowable visit day combinations. Visits have to be at the same days and times for consecutive weeks during the episode of care.

**Dynamics:** The problem is dynamic in that there are many acceptance/rejection decisions during the planning horizon. Thus, the solution depends on our scenarios. At each stage (a request arrives), decisions are whether or not the request is accepted, and if so, which day combination and time slot weekly visits should be assigned to. Patients that cannot be scheduled are rejected. We assume that the acceptance/reject decision has to be made straight away (e.g. while the patient is still on the phone) and if we reject a patient, the patient will turn to another homecare company.

**Constraints:**

- Let $i$ and $j$ be two consecutive appointments on a day, and let $g_i$ and $g_j$ represent locations of the patients assigned to those appointments. Every route for that day is feasible, if and only if

  $$K_i + sd_i + m(g_i, g_j) \leq K_j$$

  for any two consecutive appointments, $i$ and $j$.

- A task, representing a duty at a patient's home, has to be carried out as often as determined by its frequency and episode.

- One of the possible weekly visit day combinations can be selected for each patient.

- Patients, if accepted, must be serviced at same days and times every week during their service horizon.
- A nurse starts a tour from his or her home and ends the tour at his or her home again within the shift's time window.
- A nurse has to handle a task in its scheduled time period.

**Objective:** The objective is maximisation of patient visits during the planning horizon. This is different from maximisation of the number of patients served since patients need different numbers of visits. If $T$ represents a set of patients accepted over the planning horizon, our objective is:

$$\max_{T} \sum_{t \in T} f_t e c_t.$$

## 3.2 Distance and Capacity Heuristics

### 3.2.1 Distance Heuristic (DH)

The distance heuristic [Bennett and Erera, 2011] is a greedy method which assigns a new request between the pair of patients with the smallest insertion cost/additional travel time. The cost is calculated by subtracting the distance between the predecessor and successor of a request from the sum of distances between the request and its predecessor and successor. If the distance between a request and its predecessor and successor are represented as $k_1$ and $k_2$ and the distance between its predecessor and successor is $k_3$, the insertion cost, $C$ is calculated as:

$$C = k_1 + k_2 - k_3.$$

Therefore, whenever a new patient arrives to the system, the algorithm calculates the cost of insertion of that patient between all pairs of requests assigned already consecutively in each day of the week if intervals are feasible. After that, the method selects the cheapest interval in a day/days according to visit frequency of the patient.

Finally, all visits are scheduled to those cheapest days and time slots during the service horizon of the patient. The appointment time is set according to proximity of the request to its predecessor or successor. If the distance between the request and its predecessor is shorter than the distance between the request and its successor, the visit will start immediately after its predecessor visit and enough travel time of course. Otherwise, the visit starts immediately before its successor by considering service duration and travel time. If there are some days which have the same insertion costs, as a tie-breaker, we assign the visit to the day where fewer patient visits are already scheduled to balance the workload of days.

### 3.2.2 Capacity Heuristic (CH)

The distance heuristic schedules appointments next to each other, even if the travel time from one appointment to the next requires more than one time slot. In such cases it may be beneficial to allow for a longer time gap between appointments, so that future patients can be inserted in between, without requiring additional travel time.

The capacity based heuristic [Bennett and Erera, 2011] avoids scheduling a new patient directly adjacent to an existing patient if the travel time is larger than a time slot. If a new patient is more than one time slot away from other patients in the schedule, the capacity heuristic assigns his or her visit to a time slot which is far away from predecessor and successor patients to be able to assign a future request between them. Based on the example from Bennett and Erera [Bennett and Erera, 2011], let us assume that the distance between a new request and its predecessor (8.00 am) and successor (11.00 am) are 19 and 24 minutes respectively, and service time is 30 minutes for each one. Thus, candidate time slots are 9.00, 9.15, 9.30, 9.45, and 10.00 under consideration of 15-minute time intervals. If we use the distance heuristic, the request is assigned to 9.00 am. In this case, we can assign at most one additional request to 9.45, 10.00, or 10.15 if we assume that travelling between two visits takes at least a time slot (we ignore the situation in that two patients live at the same flat or similar). On the other hand, if we assign the request to 9.30, there is a possibility to assign two more patients to 8.45 and 10.15 if they need only one time slot for travelling between their predecessors and successors. Therefore, the capacity heuristic ensures to use this time slot to create gaps for suitable future patients. Of course, there must be enough space between predecessor and successor patients to put the current request into a suitable time slot. If not, requests are assigned like they are assigned with the distance heuristic.

## 3.3 Scenario Based Approach

As mentioned in previous sections, the distance and capacity heuristics are greedy algorithms which try to choose the best movement whenever a new request arrives without considering or only partially considering future requests. These heuristics accept all requests and ignore that to reject a request now can allow to accept more requests in the future. Therefore, with SBA, we try to answer two questions. First, do we accept or reject the request? And if we decide to accept the request, which time slot should weekly visits be assigned?

The basic idea behind the algorithm is to run a number of simulations (scenarios) and to see how many times the request which we have to decide on is assigned among all requests and in which time slot visits are scheduled frequently. A scenario includes a number of randomly generated requests in terms of the expected weekly demand and number of visits as can be seen in the simulation set-up in Section 3.4. We try to make a daily tour with randomly generated requests, previously accepted ones, and the current one by using the cheapest insertion heuristic whose aim is to find the shortest sub-tour. After the tour is full or all requests in the scenario are assigned, we look whether the current request has been scheduled and, if so, the time slot.

We study two different variants for SBA. First, the Daily Scenario Based Approach (DSBA) simply constructs daily tours based on daily demand and independent of a request's multiple visits in the week. Next, the Weekly Scenario Based Approach (WSBA) constructs weekly tours based on all expected weekly visits of the current patient and randomly generated requests in the scenario.

### 3.3.1 Daily Scenario Based Approach (DSBA)

In DSBA, each day in a week is evaluated separately and independently of other days in the week. Let us illustrate DSBA with an example. Assume that a new request arrives on Monday from a random location in the service area with 3-visit-per-week frequency. Episode of care and service duration do not matter since they are assumed to be the same for all patients. Now we have to decide whether we accept or reject the request.

First, we generate several scenarios for each day of the next week. Each scenario has a number of randomly generated requests and the current request as shown in Figure 3.1. To find how many requests we need to generate randomly, we calculate the average weekly demand. If we are looking at next Monday and the expected demand until that day is 10 new patient requests, the total number of visits for next week equals 25 (10*2.5), where 2.5 is the expected weekly visit frequency for a patient. We divide the total number of weekly visits by 5 to determine the average number of visits for a day. It means that 5 requests are generated for each scenario and the current request is added to them. Note that we always calculate a week of demand no matter when a request arrives as explained at the end of this section.

Next, we try to construct a tour by using requests in the scenario and patients already assigned for that day as illustrated in Figure 3.1. Requests are assigned to the tour by using the cheapest insertion heuristic until the tour is full or all requests in the scenario have been scheduled. The cheapest insertion heuristic (CIH) calculates the cost of all possible insertions and finds the one that has the lowest cost.

Once all the requests have been scheduled or no further request can be inserted, we check whether the current request has been scheduled and if so, in which time slot the visit has been scheduled. After all scenario simulations finish, we find how many times it has been accepted and which time slot it has been assigned to most frequently that day as seen on bottom right Figure 3.1. To decide which day combination (Monday-Wednesday-Friday, Tuesday-Thursday-Friday, etc.) weekly visits are scheduled, we pick up the best one, two or three days in terms of number of assignments over all scenarios. If the request cannot be scheduled for the number of days that he or she needs weekly, he or she is rejected. Algorithm 3.1 shows the pseudo code for DSBA. "nReqInTour" in Algorithm 3.1 represents how many times the request has been scheduled over all scenarios. If she or he has been assigned at least once, which is called *threshold*, we accept that request. One can see how different thresholds affect the results in Section 3.4.1.2. The number of scenarios is represented by "n" and how to determine the quantity is explained in Section 3.4.1.1.

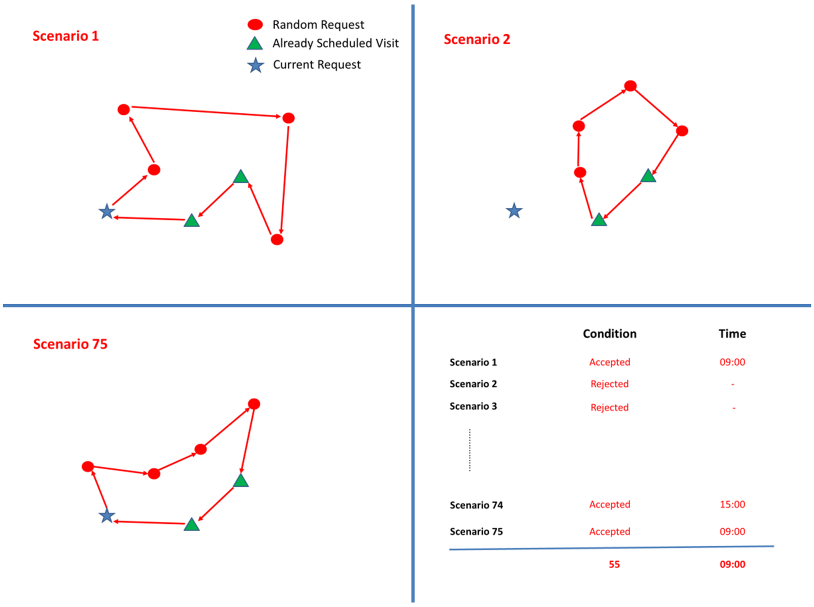



Figure 3.1: Illustration of generating scenarios and finding the number of acceptance over all scenarios and the most frequent time slot the request is assigned to.

We generate random requests based on a week of demand. For example, if a patient arrives on Wednesday, we consider a week demand when checking the next Monday or Friday. However, we have two working days until Monday and seven working days until Friday. The reason to this assumption is that requests that arrive through the end of the week are most likely accepted if the true demand is considered since no other random requests are generated due to the lack of demand. According to our experiments, this set-up outperforms the previous one if the service horizon for patients is only one week. However, if the service horizon is 4-weeks as in our case, the number of daily visits dramatically decreases because accepted patients at the end of the week start blocking acceptance of more suitable requests arriving in subsequent weeks. Therefore, we use a week of demand in all our experiments.

**Algorithm 3.1** Daily Scenario Based Approach **CIH:** Cheapest Insertion Heuristic

```
TimeSlot ← ∅
nReqInTour ← 0
for i= 1 To n  do
    ScenarioSize ← DailyVisits
    Scenario + = CurrentRequest
    for j= 1 To ScenarioSize  do
        Scenario + = RandomlyGeneratedVisits
    end for
    Tour ← Existing Visits
    while Tour is feasible and Scenario is not empty do
        MinCost ← ∞
        for k= 1 To ScenarioSize+1  do
            Cost ← CIH(Request[k])
            if Cost =< MinCost then
                MinCost ← Cost
                index ← k
            end if
        end for
        if Scenario[index] is feasible for Tour then
            Tour + = Scenario[index]
            Remove Request[index] from Scenario
        end if
    end while
```

```
    if CurrentRequest is in Tour then
        nReqInTour++
        TimeSlot + = CurrentRequestScheduleTime
    end if
end for
if nReqInTour>0 then
    Accept Patient
    VisitTime ← MostFrequentTime (TimeSlot)
end if
```

### 3.3.2 Weeky Scenario Based Approach (WSBA)

As explained in the previous section, when generating scenarios for each day in a week, different visits of the same request are considered separately for each day in DSBA. However, each request in a scenario can need 1, 2 or 3 visits in a week and this must be considered when generating scenarios for each day of the week. This is a more realistic approach since each request mimics a future patient that mostly needs multiple weekly visits. Therefore, we develop a Weekly Scenario Based Approach (WSBA) which constructs tours by taking into account all visits of requests simultaneously in each scenario. In this approach, we generate visits based on weekly demand and expected weekly visit frequency of patients, and construct a weekly schedule with corresponding daily tours by using the cheapest insertion heuristic until the tour is full or all requests in the scenario have been scheduled. After repeating the same process for several scenarios, we choose the day combination far which the current request is the most assigned over all scenarios. Patients who cannot be scheduled in any scenario are rejected. Algorithm 3.2 shows the pseudo code for WSBA. "nCombinations" represents the days for which visits of a patient can be scheduled. As an illustration, assume that there are 5 randomly generated requests (R1 to R5) with different weekly visits and Request A which is under consideration whether to accept or not in the scenario. Table 3.1 shows these requests with the number of visits they need and insertion costs in terms of travel times for each day. The insertion cost for each day is calculated as we do in DSBA. The algorithm selects the cheapest day/days depending on the number of visits that a request needs. Summing up the cost of those days gives the total cost. Monday and Wednesday have the cheapest total cost for R4 while R1 should be assigned to Monday since it needs only one visit and the cheapest insertion cost comes with Monday. Lines 15-19 in Algorithm 3.2 show the calculation of the cheapest day set as shown in the example above. Table 3.2 shows iterations where the algorithm compares requests in the scenario and selects the cheapest in terms of the average cost. The point to calculate average cost is to be able to compare insertion costs of patients who need different numbers of visits. R2 is chosen and removed from the scenario at the first iteration in Table 3.2. At the second iteration, the total costs for all remaining requests are recalculated as in Table 3.1 and Request A is selected and removed from the scenario this time due to its average cost. These iterations last until no request remains in the scenario or the tour becomes full. As can be seen in the next section, we use two different day sets, day set 1 and 2. The former covers all possible day combinations and the latter includes specific day combinations. When testing WSBA, we employ day set 2 ("nCombinations" in Algorithm 3.2) since the computational time is linear with the possible number of day combinations.

**Algorithm 3.2** Weekly Scenario Based Approach **CIH:** Cheapest Insertion Heuristic, **M:** A large positive constant

```
1:  TimeSlot ← ∅
2:  nReqInTour ← 0
3:  for i= 1 To n  do
4:      ScenarioSize ← WeeklyDemand
5:      Scenario + = CurrentRequest
6:      for j= 1 To ScenarioSize  do
7:          Scenario + = RandomlyGeneratedRequest
8:      end for
9:      Tour ← Existing Visits
10:     while Tour is feasible and Scenario is not empty do
```

```
        MinGlobalCost ← ∞
        for k= 1 To ScenarioSize+1  do
            AverageCost ← 0
            MinWeekCost ← ∞
            for p= 1 To nCombinations  do
                if CIH(Request[k],p)=<MinWeekCost then
                    MinWeekCost←CIH(Request[k],p)
                end if
            end for
            AverageCost ← MinWeekCost/Frequency[k]
            if AverageCost =< MinGlobalCost then
                MinGlobalCost←AverageCost
                index←k
            end if
        end for
        if Scenario[index] is feasible for Tour then
            Tour + = Scenario[index]
            Remove Request[index] from Scenario
        end if
    end while
    if CurrentRequest in Tour then
        nReqInTour++
        TimeSlot ← CurrentRequestScheduleTime
    end if
end for
if nReqInTour>0 then
    Accept Patient
    VisitTime ← MostFrequentTime (TimeSlot)
end if
```

## 3.4 Simulation and Results

### 3.4.1 Experimental Set-up

We run 30 simulations for each experiment. Each simulation horizon is 360 working days where each day lasts 510 minutes. A day is composed of 35 time slots. Duration between two time slots is 15 minutes. A nurse works between 08.00 and 16.30 each day during the planning horizon. Overtime and weekend work are not considered in our model. 20 days warm-up period is set at the beginning of each replication. Interarrival times between requests are exponentially distributed with mean 510, 340, or 255 minutes (we have three trials). Each patient has to be serviced 4 weeks with stochastic visit frequency 1, 2, or 3 visits per week with probabilities 0.05, 0.35, and 0.60, respectively. The first visit starts the following week after the request is accepted. Visit durations are deterministic and take 30 minutes. Each arriving customer request and randomly generated requests in scenarios uniformly arise from a small square geographic region subdivided into 900 equally-sized square subregions and a large square geographic region subdivided into 3,600 equally-sized square subregions. The reason of using two different area sizes is to observe how algorithms react to short and long travel times. Simulation parameters are shown in Table 3.3. The nurse (depot) is located in the centre of both regions. To understand differences between simulation results, we conduct independent samples t-tests and calculate p-values for each pair. Because we conduct t-tests for all experiments in this study, we want to give a clear example about how to make tests. Table 3.4 shows average daily visits of DH, CH, and DSBA by using day set 1, small service area, and 340-minute interarrival time for each simulation. Our null hypothesis is that the average daily visit of DSBA is equal to the average daily visits of DH and CH. Therefore, the alternative hypothesis is that the average daily visit of DSBA is different from the other two methods. We use Microsoft Excel for tests. To be able to conduct independent t-tests in Excel, we must determine whether to use one tail or two tails and whether variances are equal or not. Although we reduce variances by common

Table 3.1: Assignment cost for each visit of requests and total cost

| | Visit | Monday | Tuesday | Wednesday | Thursday | Friday | Day set | Total cost |
|---|---|---|---|---|---|---|---|---|
| R1 | 1 | 50 | 60 | 55 | 80 | 80 | Mon | 50 |
| R2 | 3 | 30 | ... | 40 | ... | 20 | Mon-Wed-Fri | 90 |
| R3 | 3 | 50 | ... | 30 | ... | 40 | Mon-Wed-Fri | 120 |
| R4 | 2 | 50 | 60 | 50 | 80 | 60 | Mon-Wed | 100 |
| R5 | 2 | 80 | 40 | 50 | 40 | 70 | Tue-Thu | 80 |
| A | 3 | 70 | ... | 50 | ... | 70 | Mon-Wed-Fri | 190 |

Table 3.2: Selection of requests

| | Iteration 1 | | | Iteration 2 | | Iteration 3 | |
|---|---|---|---|---|---|---|---|
| | Visit | Total cost | Average cost | Total cost | Average cost | Total cost | Average cost |
| R1 | 1 | 50 | 50 | 60 | 60 | 30 | **30** |
| R2 | 3 | 90 | **30** | ... | ... | ... | ... |
| R3 | 3 | 120 | 40 | 150 | 50 | 120 | 40 |
| R4 | 2 | 100 | 50 | 120 | 60 | 140 | 70 |
| R5 | 2 | 80 | 40 | 150 | 50 | 100 | 50 |
| A | 3 | 190 | 63 | 100 | **33** | ... | ... |

random numbers for each test, it is better to conduct F test to check whether variances are equal. According to Table 3.5, all p values of the F test in Excel are greater than the threshold value, 0.05. It means that variances are not statistically different. After we make sure that samples have equal variances, we can conduct independent samples t tests with two tails and for equal variances. Table 3.6 shows p values for t-tests. As can be seen, values are much lower than the threshold value. Thus, we can reject the null hypothesis and say average daily visits of the three methods are statistically different. Instead of giving p values for all tests, statistically different results are written in bold font in tables. We have two different set-ups for visit days each patient can be assigned to according to his weekly visit frequency. In the first set-up, each patient can be scheduled any combination of days in the week. Because we do not allow weekend work, there are $\binom{n}{f}$ day combinations for a patient with $f$ representing the visit frequency and $n$ representing the number of days (Monday, Tuesday, Wednesday, Thursday, Friday). This is called day set 1. Although most studies in the literature do not mention to employ special visit day combinations when assigning requests, some authors [Duque et al., 2015] emphasize not to use sequential days if multiple visits are taken into consideration. And, it does not make sense to perform some tasks such as cooking, bathing, etc. the first two or three days at the beginning of a week and to do nothing at the remaining days when considering real life cases. Thus, we also use another day set-up which does not allow to schedule sequential days when the visit frequency of a patient are two or three. Therefore, only the following visit day combinations can be assigned to a patient who needs two visits in a week, *((Monday,Friday),(Monday,Thursday),(Monday,Wednesday),(Tuesday,Friday),(Tuesday,Thursday))* and a patient who needs three visits in a week, *(Monday,Wednesday,Friday)*. This set-up is called day set 2.

#### 3.4.1.1 Determination of Scenario Size

In DSBA and WSBA, we fixed the scenario size to 75. Obviously, a large number of scenarios means longer computational time. On the contrary, a lower size of scenarios

Table 3.3: Simulation Setup

| Simulation Parameters | |
|---|---|
| Simulation Horizon (day) | 360 |
| Warm-up Period (day) | 20 |
| Daily Working Time (minute) | 510 |
| Service Horizon (week) | 4 |
| Interarrival Times (minute) | 510,340,255 |
| Weekly Visit Frequency | 1,2,3 |
| Weekly Visit Probability | 0.05,0.35,0.60 |
| Small Area | $X \in [0, 30]$ and $Y \in [0, 30]$ |
| Large Area | $X \in [0, 60]$ and $Y \in [0, 60]$ |

can cause decreasing quality of estimation for appointment times. Therefore, we tried different numbers of scenarios to observe how it affects results. Figure 3.2 shows the average number of daily visits under different scenario sizes and interarrival times for day set 1, a small region, and the predefined experimental setting. The results for the three different interarrival times stabilise at scenario sizes above 70 or 80. Although there are other peaks when sizes are between 130-150, it is hard to fix a number for different interarrival times and computational cost significantly increases around these points.

Table 3.4: Average daily visits of DH, CH, and DSBA by using day set 1, small service area, and 340-minute interarrival time for each simulation

| Experiment | SBA | DH | CH | Experiment | SBA | DH | CH |
|---|---|---|---|---|---|---|---|
| 1 | 9.36 | 9.07 | 9.17 | 16 | 9.46 | 9.09 | 9.08 |
| 2 | 9.46 | 9.08 | 9.07 | 17 | 9.25 | 9.03 | 9.22 |
| 3 | 9.52 | 9.12 | 9.30 | 18 | 9.04 | 9.05 | 9.28 |
| 4 | 9.39 | 9.31 | 9.41 | 19 | 9.35 | 9.25 | 9.23 |
| 5 | 9.03 | 9.10 | 8.94 | 20 | 9.49 | 9.15 | 8.99 |
| 6 | 9.07 | 9.11 | 9.13 | 21 | 9.23 | 9.07 | 9.20 |
| 7 | 9.52 | 9.17 | 9.33 | 22 | 9.39 | 9.17 | 9.08 |
| 8 | 9.29 | 8.91 | 9.10 | 23 | 9.40 | 8.72 | 9.05 |
| 9 | 9.42 | 9.08 | 9.00 | 24 | 9.50 | 8.83 | 9.16 |
| 10 | 9.13 | 8.90 | 9.19 | 25 | 9.54 | 9.10 | 9.11 |
| 11 | 9.51 | 8.95 | 9.24 | 26 | 9.49 | 8.85 | 9.17 |
| 12 | 9.57 | 8.92 | 9.03 | 27 | 9.49 | 8.97 | 9.36 |
| 13 | 9.32 | 8.97 | 8.72 | 28 | 9.44 | 8.99 | 9.18 |
| 14 | 9.38 | 9.00 | 9.20 | 29 | 9.51 | 8.93 | 9.22 |
| 15 | 9.41 | 8.87 | 9.19 | 30 | 9.47 | 9.19 | 8.92 |
| | | | | **Average** | **9.36** | **9.04** | **9.13** |

Table 3.5: F tests

| | DSBA | DH | CH |
|---|---|---|---|
| **DSBA** | ... | 0.45 | 0.75 |
| **DH** | 0.45 | ... | 0.66 |
| **CH** | 0.75 | 0.66 | ... |

Table 3.6: Independent sample t-tests

| | DSBA | DH | CH |
|---|---|---|---|
| **DSBA** | ... | 1.75E-13 | 4.57E-08 |
| **DH** | 1.75E-13 | ... | 2.85E-03 |
| **CH** | 4.57E-08 | 2.85E-03 | ... |

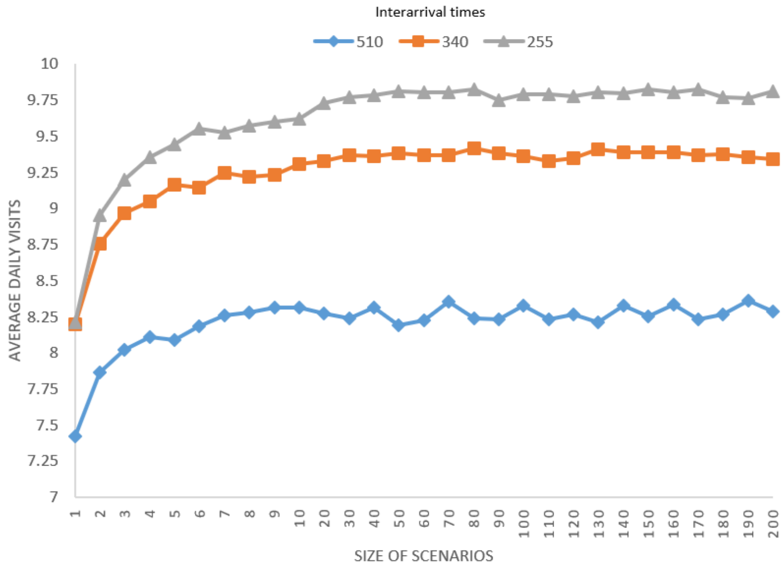



Figure 3.2: Average daily visits under different scenario sizes and interarrival times

#### 3.4.1.2 Determination of Acceptance Threshold

As we mentioned previously, one of our aims in this study is to develop an acceptance policy. We believe that rejection of some patients helps to accept more patients in the future. In DSBA and WSBA, some scenarios are generated and daily/weekly tours are constructed. The purpose is to check whether or not the current request is accepted. However, how many times across the number of scenarios should a patient be assigned to be able to accept it? To determine the setting, we tried different acceptance thresholds as in Figure 3.3. Again we constructed three trials for different interarrival times and same experimental setting as we do in Section 3.4.1.1. Figure 3.3 clearly demonstrates that average daily visits tend to reduce when the acceptance threshold is increased. The reason is that accepting a request is getting harder when we increase the threshold. Particularly, if the demand is high, the decline of average number of daily visits is sharper since high scenario sizes and threshold decrease the probability of acceptance. Therefore, we fixed the acceptance threshold to 1. It means that we accept a patient if he or she can be scheduled at least once over 75 scenarios. Note that always accepting the patient would be similar

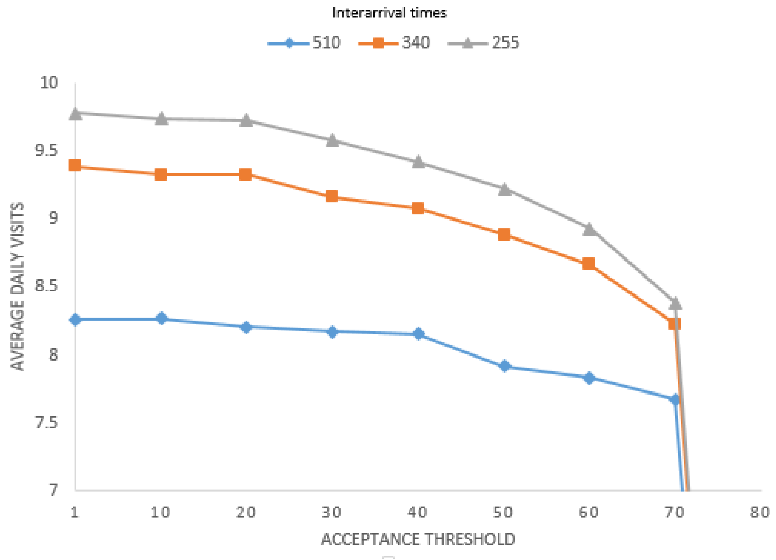



Figure 3.3: Average daily visits for different acceptance thresholds

to DH and leads to inferior results.

#### 3.4.1.3 Demand Fluctuations

When generating random requests for each scenario, we calculate the weekly expected demand based on interarrival times and daily visits based on expected visit frequency for each patient as mentioned in Section 3.3.1. However, it is possible that realized demands might be lower or higher than expected. In this section, we test how robust our algorithm is against demand fluctuations.

According to interarrival times, weekly visit probability, and frequency we employ in our tests, the realized number of randomly generated visits for each scenario can be 5, 4, and 3 for 255, 340, and 510 minutes interarrival times, respectively. In tests, we generate lower or higher number of random visits for each scenario independently from the current interarrival time when patients arrive according to predefined interarrival time during the simulation horizon. After 30 simulations, we calculate average daily visits under the lower or higher number of random visits for scenarios. We also provide average daily visits of DH and CH as a benchmark. It is clear that DH and CH are not effected on demand fluctuations since they do not make any future estimation when assigning visits now.

Table 3.7: Average daily visits of SBA under demand fluctuations in a small area

| **Visits** | **255** | **340** | **Visits** | **510** |
|---|---|---|---|---|
| 2 | 9.56 | 9.29 | 1 | 8.29 |
| 3 | 9.68 | 9.33 | 2 | 8.29 |
| 4 | 9.77 | **9.38** | 3 | **8.31** |
| 5 | **9.79** | 9.40 | 4 | 8.31 |
| 6 | 9.83 | 9.38 | 5 | 8.30 |
| 7 | 9.79 | 9.36 | 6 | 8.19 |
| **DH** | **9.28** | **9.03** | **DH** | **8.19** |
| **CH** | **9.49** | **9.14** | **CH** | **8.21** |

Table 3.8: Average daily visits of SBA under demand fluctuation in a large area

| **Visits** | **255** | **340** | **Visits** | **510** |
|---|---|---|---|---|
| 2 | 8.03 | 7.73 | 1 | 7.01 |
| 3 | 8.10 | 7.80 | 2 | 7.09 |
| 4 | 8.18 | **7.88** | 3 | **7.11** |
| 5 | **8.26** | 7.88 | 4 | 7.09 |
| 6 | 8.27 | 7.88 | 5 | 7.07 |
| 7 | 8.27 | 7.85 | 6 | 7.05 |
| **DH** | **7.79** | **7.54** | **DH** | **6.97** |
| **CH** | **7.46** | **7.18** | **CH** | **6.57** |

Tables 3.7 and 3.8 show average daily visits of SBA under demand fluctuation and average daily visits of DH and CH in small and large areas. Bold numbers represent average daily visits of SBA when the number of randomly generated visits are identical to the expected number of visits. Particularly, if the demand is realised lower than it is expected, average daily visits decrease more than when the demand

is realised higher than it is expected. Furthermore, SBA provides maximum number of visits when the number of randomly generated requests are one more than they are supposed to be. Although we can change the algorithm to generate one request more for each scenario, we prefer not to do that and continue our experiments with expected number of visits. The most important conclusion derived from results is that SBA provides higher average daily visits compared to DH and CH even though weekly demands are estimated higher or lower than real demands.

#### 3.4.1.4 Sampling Methodology

One of the most important parts of our methodology is to generate scenarios to be able to predict future patient requests. When generating scenarios, we produce patient requests from different locations in the service area. A patient location is defined as $(X, Y)$ in the service area and both $X$ and $Y$ are uniformly distributed. We use Monte Carlo sampling to generate locations of patients independently. Because it relies on pure randomness, we end up with some locations clustered closely, while other regions within the service area get no samples. This could affect acceptance or rejection decisions since if the patient location is far away from the region in which random requests are clustered, this is always possible not to be integrated into the tour of nurse. We have to make sure that the Monte Carlo sampling method should not impact results significantly. Therefore, we employ Latin Hypercube sampling to see whether or not another sampling method can change results. Latin Hypercube sampling targets to expand the sample points more evenly across all possible values [McKay et al., 1979]. In our case, it ensures that generated requests in each scenario are not clustered in a subregion.

Figure 3.4 shows step by step how to generate requests in the service region with Latin Hypercube sampling. In this example, we generate three requests for a scenario. In step 1, the method divides the service region into nine equal subregions. The number of subregions depends on the number of patient requests in the scenario. For example, if we have four requests, the method creates 16 subregions. Because

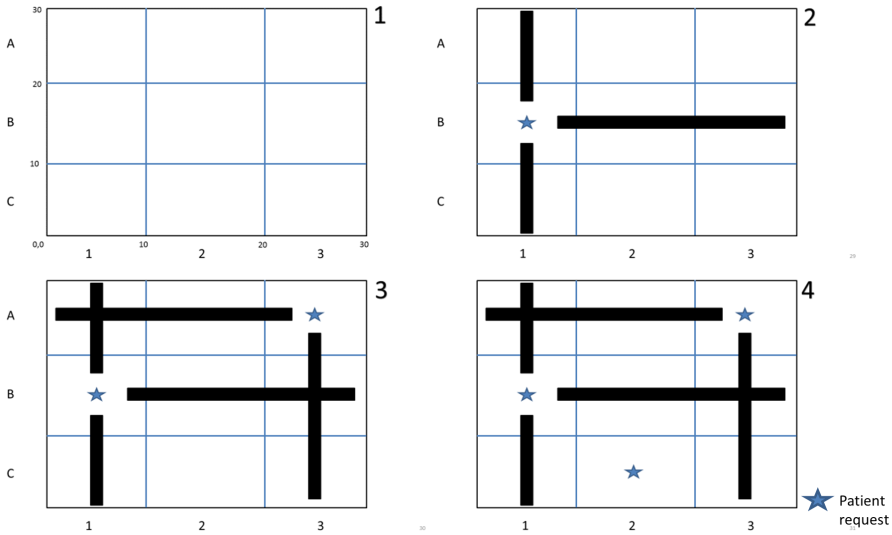



Figure 3.4: Request generation with Latin Hypercube sampling method

we assume that the service area is a square, each subregion can be defined by the corresponding column and row. For example, the subregion located at the centre of service area is called *B2*. In step 2, the method randomly chooses a column and row, *B1*, among all columns and rows. Location of the first request is randomly assigned from *B1* subregion. After that, the method eliminates column *1* and row *B*. In step 3, the method randomly chooses a column and row among the remaining columns and rows. Now we have row *A* and *C*, and column *2* and *3*. Location of the second request is assigned from *A3* subregion. Again, the method eliminates column *3* and row *A*. In the last step, location of the third request is assigned from *C2* subregion since that is the only remaining subregion.

Table 3.9 shows results based on different sampling methods for a vary of interarrival times and service areas. It is clear that there is no significant difference between results of both methods. However, samplings based on Latin Hypercube method result in slightly lower average daily visits and acceptance rates, and longer travel times compared to samplings based on Monte Carlo method. The reason might be that we generate actual patient arrivals based on Monte Carlo method while generating patient requests in scenarios based on Latin Hypercube method. It

Table 3.9: Average daily visits, travel times per visit, and acceptance rates based on scenario generations with Latin Hypercube sampling and Monte Carlo method (Random)

| | | Average daily visit | | Travel time per visit | | Acceptance rate | |
|---|---|---|---|---|---|---|---|
| **Area size** | **Interarrivals** | **HyberCube** | **Random** | **HyberCube** | **Random** | **HyberCube** | **Random** |
| | **510** | 8.35 | 8.31 | 16.28 | 15.46 | 0.84 | 0.83 |
| **Small** | **340** | 9.34 | 9.38 | 15.07 | 14.68 | 0.64 | 0.65 |
| | **255** | 9.72 | 9.79 | 14.29 | 13.68 | 0.52 | 0.53 |
| | **510** | 7.08 | 7.09 | 27.05 | 25.87 | 0.72 | 0.72 |
| **Large** | **340** | 7.81 | 7.88 | 24.73 | 24.42 | 0.54 | 0.55 |
| | **255** | 8.24 | 8.26 | 22.74 | 22.63 | 0.44 | 0.45 |

is hard to generate actual patient arrivals based on Latin Hypercube method since we do not know how many patients arrive during the planning horizon. Because results of both methods are not statistically different, we use random assignments for our all experiments.

### 3.4.2 WSBA and DSBA

In this section, we compare the two different solution methodologies which we developed, WSBA and DSBA. As explained in the previous section, the main difference between the two methodologies is that each tour constructed for a day is independent of the remaining days in the week in DSBA. On the other hand, weekly tours are constructed by using weekly visits belonging to same requests in WSBA. The latter is closer to our problem setting since requests need one, two, or three visits in a week and generating different requests for each scenario without considering these visits as in DSBA can affect the results. However, Figures 3.5, 3.6, 3.7, and Tables 3.10 and 3.11 show that results of average daily visits, travel times per visit, and acceptance rates are close to each other. It is hard to say whether one is superior to the other since DSBA provides slightly better results in some cases while there are other cases where WSBA works well. Because our objective is to maximise average daily visits, it is more important to look at results of visits for WSBA and DSBA. As can be seen in Figure 3.5 and the first three rows in Table 3.10, DSBA results are slightly higher for the small region, but the only difference between the average number of visits for WSBA and DSBA under large area and high demand scenario is statistically significant in favour of WSBA.

Computational cost is a crucial factor for this study since the decision has to be made as soon as someone requests for the service. In other words, faster decision making means happier customers. Therefore, execution times are measured for WSBA, DSBA, DH, and CH as in Table 3.12. Each time is measured during a-year simulation horizon in which day set 2 and a small area are considered. Although execution times for DSBA and WSBA seem relatively long compared to DH and CH, the execution time for a patient's acceptance and assignment decision lasts less than a second for the longest case as shown in Table 3.13. Execution times for WSBA is significantly longer than DSBA execution times even though we use day set 2 for the trial. It is clear that assessing a whole week with all visits of different requests in WSBA significantly increases computations compared to decomposing a week into separate days and evaluating them independently in DSBA. We decided to use DSBA since results explained above do not show a large difference and computational cost of DSBA is much lower than WSBA's.

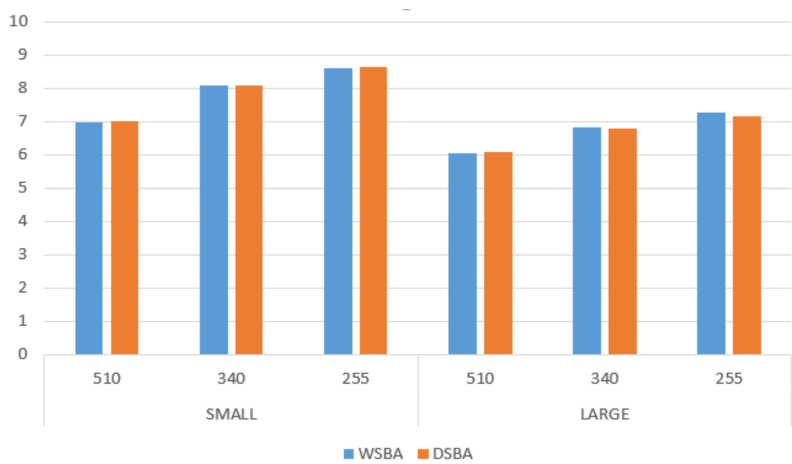



Figure 3.5: Average daily visits for WSBA and DSBA

### 3.4.3 DSBA, Distance, and Capacity Heuristics

Table 3.14 shows average daily visits according to DH, CH, and DSBA. As one can see in the tables, our methodology gives superior results for both small and large regions and different interarrival times. Particularly, daily visits increase substantially compared to DH and CH in a small region if demand is relatively high. In a large area, average daily visits by using DH is higher than by using CH, but the improvement by SBA reaches around 11% and 6% compared to CH and DH. If 0.05 is selected as a threshold for t test, it can be seen that all improvements are statistically significant.

Table 3.15 demonstrates travel times per visit for the three approaches. DH and

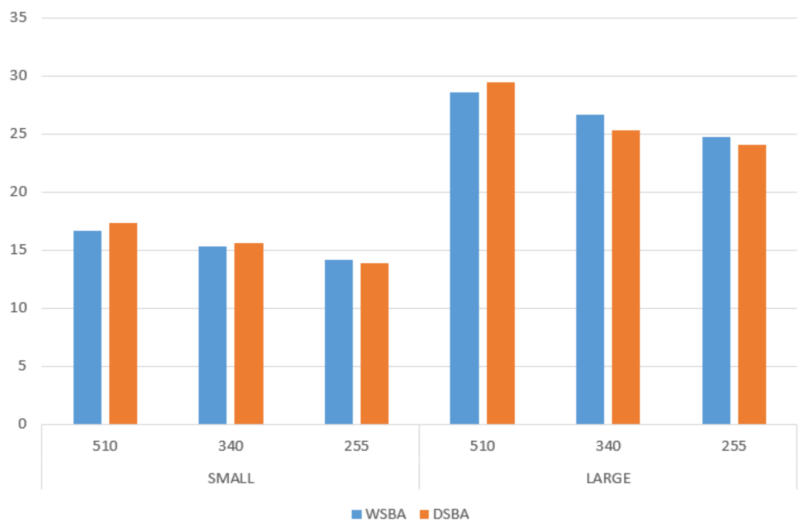



Figure 3.6: Travel times per visit for WSBA and DSBA (minutes)

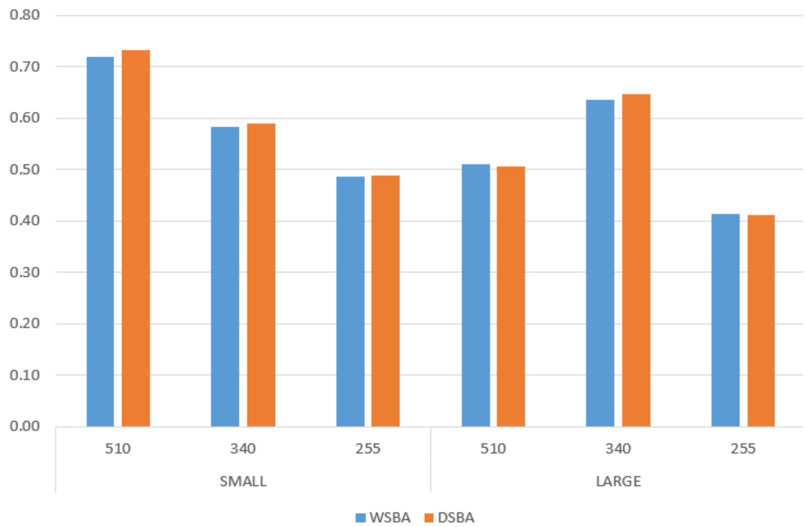



Figure 3.7: Acceptance rates for WSBA and DSBA

Table 3.10: Comparisons of WSBA and DSBA in terms of average daily visits, travel times per visit, and acceptance rates for the small region

| | | **WSBA** | **DSBA** | p value |
|---|---|---|---|---|
| Daily Visits | 510 | 6.97 | 7.00 | 0.282 |
| | 340 | 8.07 | 8.09 | 0.648 |
| | 255 | 8.61 | 8.65 | 0.282 |
| Travel Times | 510 | **16.67** | 17.36 | 0.005 |
| | 340 | **15.35** | 15.64 | 0.017 |
| | 255 | 14.18 | **13.92** | 0.013 |
| Acceptance Rate | 510 | 0.72 | **0.73** | 0.039 |
| | 340 | 0.58 | 0.59 | 0.173 |
| | 255 | 0.49 | 0.49 | 0.535 |

Table 3.11: Comparisons of WSBA and DSBA in terms of average daily visits, travel times per visit, and acceptance rates for the large region

| | | **WSBA** | **DSBA** | p value |
|---|---|---|---|---|
| Daily Visits | 510 | 6.05 | 6.08 | 0.499 |
| | 340 | 6.81 | 6.81 | 0.865 |
| | 255 | **7.28** | 7.18 | 0.001 |
| Travel Times | 510 | **28.58** | 29.43 | 0.005 |
| | 340 | 26.66 | **25.34** | 0.001 |
| | 255 | 24.75 | **24.03** | 0.005 |
| Acceptance Rate | 510 | 0.51 | 0.51 | 0.067 |
| | 340 | 0.64 | 0.65 | 0.202 |
| | 255 | 0.41 | 0.41 | 0.338 |

Table 3.12: Execution times for each method (milliseconds) in a-year simulation horizon

| Method | 510 | 340 | 255 |
|---|---|---|---|
| **WSBA** | 24927 | 78676 | 177489 |
| **DSBA** | 1741 | 2813 | 6873 |
| **CH** | 33 | 47 | 56 |
| **DH** | 32 | 42 | 51 |

Table 3.13: Execution times for each method (milliseconds) for a patient's acceptance and assignment decision

| Method | 510 | 340 | 255 |
|---|---|---|---|
| **WSBA** | 77.90 | 163.91 | 277.33 |
| **DSBA** | 5.44 | 5.86 | 10.74 |
| **CH** | 0.10 | 0.10 | 0.09 |
| **DH** | 0.10 | 0.09 | 0.08 |

Table 3.14: Average daily visits for DH, CH, and DSBA by using day set 1

| Region | Times | DH | DSBA | % | CH | DSBA | % |
|---|---|---|---|---|---|---|---|
| Small | 510 | 8.19 | **8.31** | 1.46 | 8.21 | **8.31** | 1.32 |
| Small | 340 | 9.03 | **9.38** | 3.87 | 9.14 | **9.38** | 2.61 |
| Small | 255 | 9.28 | **9.79** | 5.50 | 9.49 | **9.79** | 3.22 |
| Large | 510 | 6.97 | **7.09** | 1.81 | 6.57 | **7.09** | 7.98 |
| Large | 340 | 7.54 | **7.88** | 4.49 | 7.18 | **7.88** | 9.68 |
| Large | 255 | 7.79 | **8.26** | 5.97 | 7.46 | **8.26** | 10.75 |

Table 3.15: Average travel time per visit for DH, CH, and DSBA (minutes) by using day set 1

| Region | Times | DH | DSBA | % | CH | DSBA | % |
|---|---|---|---|---|---|---|---|
| Small | 510 | **14.75** | 15.46 | 4.76 | **14.92** | 15.46 | 3.57 |
| Small | 340 | 15.24 | **14.68** | -3.72 | 15.07 | **14.68** | -2.62 |
| Small | 255 | 14.98 | **13.68** | -8.68 | 14.88 | **13.68** | -8.05 |
| Large | 510 | 26.63 | **25.87** | -2.86 | 26.83 | **25.87** | -3.58 |
| Large | 340 | 26.17 | **24.42** | -6.70 | 26.64 | **24.42** | -8.35 |
| Large | 255 | 25.75 | **22.63** | -12.11 | 26.07 | **22.63** | -13.19 |

CH provide shorter travel times than DSBA under low demand since it does not benefit from its ability to select more suitable requests. When demand is higher, DSBA also ensures travel times at least as good as DH and CH or better even though number of patients serviced is more compared to the other two methods. Particularly, travel times in SBA are significantly lower in a large area and when demand is moderate and high.

Table 3.16 represents acceptance rates (number of accepted requests/total requests) for the three methods. Although DH and CH accept all they can and do not reject any request if they have an available place for it, acceptance rates by our methodology are higher in all circumstances. This demonstrates that rejection of some requests now can help to accept more requests overall in the future. The proposed methodology takes demand fluctuation into account and is willing to accept as many patients as possible if the demand is low. However, it can be seen that our methodology significantly increases acceptance rates under scenarios of small region-high demands and large region. All the results of average daily visits, travel times per visit and acceptance rates are statistically different.

Tables 3.17 to 3.19 show average daily visits, travel times per visit, and acceptance rates for DH, CH, and DSBA for day set 2. DSBA provides higher average

Table 3.16: Acceptance rates for DH, CH, and DSBA by using day set 1

| **Region** | **Times** | **DH** | **DSBA** | % | **CH** | **DSBA** | % |
|---|---|---|---|---|---|---|---|
| Small | 510 | 0.81 | **0.83** | 2.34 | 0.82 | **0.83** | 1.73 |
| Small | 340 | 0.62 | **0.65** | 4.63 | 0.63 | **0.65** | 3.18 |
| Small | 255 | 0.48 | **0.53** | 8.50 | 0.49 | **0.53** | 7.16 |
| Large | 510 | 0.71 | **0.72** | 1.24 | 0.67 | **0.72** | 6.78 |
| Large | 340 | 0.52 | **0.55** | 5.82 | 0.50 | **0.55** | 9.68 |
| Large | 255 | 0.41 | **0.45** | 9.39 | 0.40 | **0.45** | 13.62 |

Table 3.17: Average daily visits for DH, CH, and DSBA by using day set 2

| **Region** | **Times** | **DH** | **DSBA** | % | **CH** | **DSBA** | % |
|---|---|---|---|---|---|---|---|
| Small | 510 | 6.52 | **7.00** | 7.4 | 6.63 | **7.00** | 5.6 |
| Small | 340 | 7.80 | **8.09** | 3.7 | 7.85 | **8.09** | 3.1 |
| Small | 255 | 8.29 | **8.65** | 4.3 | 8.51 | **8.65** | 1.6 |
| Large | 510 | 5.9 | **6.08** | 3.1 | 5.52 | **6.08** | 10.2 |
| Large | 340 | 6.69 | **6.81** | 1.9 | 6.32 | **6.81** | 7.8 |
| Large | 255 | 7.06 | **7.18** | 1.7 | 6.73 | **7.18** | 6.7 |

Table 3.18: Average travel time per visit for DH, CH, and DSBA (minutes) by using day set 2

| **Region** | **Times** | **DH** | **DSBA** | % | **CH** | **DSBA** | % |
|---|---|---|---|---|---|---|---|
| Small | 510 | 18.88 | **17.36** | -8.6 | 18.15 | **17.36** | -4.4 |
| Small | 340 | 16.67 | **15.64** | -6.32 | 16.82 | **15.64** | -7.5 |
| Small | 255 | 16.35 | **13.92** | -14.9 | 16.11 | **13.92** | -13.4 |
| Large | 510 | 30.95 | **29.27** | -5.4 | 32.22 | **29.27** | -9.2 |
| Large | 340 | 27.40 | **25.34** | -7.5 | 28.50 | **25.34** | -11.1 |
| Large | 255 | 28.46 | **24.03** | -15.6 | 29.23 | **24.03** | -17.8 |

Table 3.19: Acceptance rates for DH, CH, and DSBA by using day set 2

| **Region** | **Times** | **DH** | **DSBA** | % | **CH** | **DSBA** | % |
|---|---|---|---|---|---|---|---|
| Small | 510 | 0.68 | **0.73** | 7.4 | 0.71 | **0.73** | 2.8 |
| Small | 340 | 0.60 | 0.60 | 0 | 0.59 | 0.60 | 1.7 |
| Small | 255 | 0.48 | 0.49 | 2.1 | **0.51** | 0.49 | 4.1 |
| Large | 510 | 0.65 | 0.65 | 0 | 0.62 | **0.65** | 4.8 |
| Large | 340 | 0.53 | **0.54** | 1.9 | 0.50 | **0.54** | 8 |
| Large | 255 | **0.44** | 0.43 | -2.3 | 0.42 | **0.43** | 2.3 |

daily visits and lower travel times per visit for both small and large regions and all examined interarrival times. All differences of average daily visits and travel times are statistically significant. Particularly, percentages of increase for average daily visits provided by DSBA tend to increase when the demand is low. On the other hand, saving travel times per visit is going up when size of area gets bigger and demand gets higher for DSBA. However, we cannot say that acceptance rates are statistically different from each other in some cases.

If one compares average daily visits in Table 3.14 with visits in Table 3.17, one can observe that the percentage increase for average daily visit for day set 2 lessens when demand is getting higher compared to the situation for day set 1. A possible explanation is that the only day combination for a patient who needs three visits per week is *(Monday, Wednesday, Friday)*. Since 60% of patients demand three day visits, these days are quickly getting full at the high demand and SBA does not have many options to optimise remaining requests and days. Therefore, the gap between average daily visits of SBA and DH is getting smaller.

## 3.5 Patient Dependent Service Times

In the previous setting, we assume that the service time for each visit is deterministic and takes 30 minutes. However, it is highly possible that some tasks last longer or shorter than other tasks in real life. Therefore, we add this case into our model by using two different scenarios. When a patient arrives, the visit time is generated stochastically. Visit times can be 15, 30, and 45 minutes with probability 0.30, 0.35, and 0.35 in the first scenario and 0.10, 0.30, and 0.60 respectively in the second scenario. When generating random requests for DSBA, we also take patient dependent service times into consideration. For example, let us assume that we generate 10 requests for each scenario. We expect that 6 of them need 45 minute service, 3 of them need 30 minute service, and one takes 15 minute service according to the second scenario. Moreover, we test two different cost factors when assigning requests in the scenario generation phase. First, as in the previous setting, we only consider travel times when searching for the most suitable request. Second, the ratio of travel time to service time is employed as a criterion for insertion. The point of service time consideration is that acceptance of requests with longer service times can shorten overall daily travel times since the nurse must visit less patients. Of course, accepted average number of patients decreases due to longer service times, but we look at average daily service duration representing how long a nurse spent for only service purpose to be able to compare results. For example, assume that we have three requests in a scenario and their insertion costs in terms of travel time are 20, 30, and 35 minutes. If only travel time cost is taken into consideration as a selection criterion, the algorithm chooses the first request due to the lowest travel time cost. Now, assume that service time of requests is 15, 30, and 45. If we divide travel times by service times, we get 1.33, 1, and 0.78. If the ratio of travel time to service time is considered as cost, we choose the third request.

According to Table 3.20 and 3.21, there is no statistically significant differences between the two cost factors. However, when we consider the ratio of travel time to service time as cost criterion, daily service duration is always slightly higher than when we use only travel time as cost factor. However, results of both cost strategies for service durations, average daily visits, travel times per visit, and acceptance rates are very close. No matter what cost strategy is chosen, results under patient dependent service times are superior compared to DH and CH.

## 3.6 Relaxation of Visit Times

One of constraints in this study is to keep service continuity which guarantees that patients are visited at the same times during their service horizons. It is a quite common practice accepted by HHC companies and researchers. However, one may wonder how violation of service continuity affects daily visits and travel times. In other words, how should the trade off between the service continuity and daily visit/travel cost be handled if the service continuity is a soft constraint? To answer this question, we develop a new model where schedules are made weekly. According to

Table 3.20: Comparison of DSBA with two different cost factors, DH, and CH under patient dependent service times for day set 1 and the small service region

| | Interarrival time | Strategy | Duration | Visits | TravelTime | Acceptance rate |
|---|---|---|---|---|---|---|
| **Scenario 1** | **255** | **Travel/Service** | **273.75** | **10.37** | **13.92** | **0.53** |
| | | **Travel** | **271.71** | **10.44** | **13.51** | **0.52** |
| | | **DH** | 263.90 | 9.92 | 13.81 | 0.51 |
| | | **CH** | 265.80 | 10.21 | 14.64 | 0.51 |
| | **340** | **Travel/Service** | **268.40** | **9.69** | **14.70** | **0.64** |
| | | **Travel** | **267.24** | **9.69** | **14.47** | **0.64** |
| | | **DH** | 260.58 | 9.50 | 14.99 | 0.63 |
| | | **CH** | 259.50 | 9.57 | 15.03 | 0.63 |
| **Scenario 2** | **255** | **Travel/Service** | **296.29** | **8.73** | **14.62** | **0.44** |
| | | **Travel** | **295.06** | **8.78** | **14.40** | **0.45** |
| | | **DH** | 286.05 | 8.40 | 14.75 | 0.43 |
| | | **CH** | 287.05 | 8.59 | 15.62 | 0.44 |
| | **340** | **Travel/Service** | **291.06** | **8.30** | **15.48** | **0.55** |
| | | **Travel** | **289.73** | **8.28** | **15.34** | **0.55** |
| | | **DH** | 280.84 | 8.15 | 16.04 | 0.54 |
| | | **CH** | 280.93 | 8.16 | 16.09 | 0.55 |

Table 3.21: Comparison of DSBA with two different cost factors, DH, and CH under patient dependent service times for day set 2 and the small service region

| | Interarrival time | Strategy | Duration | Visits | TravelTime | Acceptance rate |
|---|---|---|---|---|---|---|
| **Scenario 1** | **255** | **Travel/Service** | **234.45** | **8.88** | **13.75** | **0.46** |
| | | **Travel** | **232.70** | **8.94** | **13.35** | **0.46** |
| | | **DH** | 223.67 | 8.41 | 13.36 | 0.45 |
| | | **CH** | 225.28 | 8.66 | 14.16 | 0.45 |
| | **340** | **Travel/Service** | **226.44** | **8.19** | **14.51** | **0.55** |
| | | **Travel** | **225.46** | **8.18** | **14.29** | **0.55** |
| | | **DH** | 213.32 | 7.78 | 13.95 | 0.53 |
| | | **CH** | 212.44 | 7.84 | 13.99 | 0.54 |
| **Scenario 2** | **255** | **Travel/Service** | **253.75** | **7.48** | **14.44** | **0.39** |
| | | **Travel** | **252.70** | **7.52** | **14.23** | **0.39** |
| | | **DH** | 242.45 | 7.12 | 14.27 | 0.37 |
| | | **CH** | 243.30 | 7.28 | 15.11 | 0.38 |
| | **340** | **Travel/Service** | **245.56** | **7.00** | **15.29** | **0.48** |
| | | **Travel** | **244.44** | **6.99** | **15.15** | **0.48** |
| | | **DH** | 229.91 | 6.67 | 15.93 | 0.46 |
| | | **CH** | 229.98 | 6.68 | 15.98 | 0.46 |

this model, schedule times of accepted patients are determined at the beginning of each week by the cheapest insertion heuristic after their first week visits have been decided when they arrive. Therefore, patients are informed of visit times at least one week before, but may not be informed of all visit times for all weeks at the start of service horizon. For example, let us assume that a patient arrives on Monday and is accepted. He or she is only informed about next week's visit times. At the beginning of next week, which can be on Monday or Sunday depending on a decision maker, all accepted patients' visits are scheduled for the following week. So if we assume that the patient arrives at week 0, his first weekly visits are scheduled in week 1 and second weekly visits are scheduled in week 2 at the beginning of week 1. Note that the first weekly visits of a patient are assigned as soon as he or she arrives since dynamic patient arrivals and fast decisions are main considerations of this research. If we made routing and scheduling decisions after collecting a number of patient requests in a period of time, this would be a static HHC problem and, as we have discussed in the Literature Review Section, there are many studies considering this problem setting.

Table 3.22: Average daily visits under strict and flexible assignments for two day sets

| | | Day Set 1 | | | Day Set 2 | | |
|---|---|---|---|---|---|---|---|
| Region | Interarrival times | Strict | Flexible | % | Strict | Flexible | % |
| Small | 510 | 8.31 | **8.85** | 6.5 | 7.00 | **7.57** | 8.1 |
| Small | 340 | 9.38 | **9.94** | 6.0 | 8.09 | **8.83** | 9.2 |
| Small | 255 | 9.79 | **10.31** | 5.3 | 8.65 | **9.43** | 9.0 |
| Large | 510 | 7.09 | **7.78** | 9.7 | 6.08 | **6.67** | 9.7 |
| Large | 340 | 7.88 | **8.49** | 7.7 | 6.81 | **7.67** | 12.6 |
| Large | 255 | 8.26 | **8.85** | 7.2 | 7.18 | **8.14** | 13.4 |

Table 3.22 shows average daily visits under strict and flexible assignments for two day sets. The strict assignment considers service continuity while schedules are

prepared at the beginning of each week as explained above in the flexible assignment. According to results, average daily visits increase between 5% and 13% when schedules are made by ignoring service continuity. Table 3.23 shows travel times per visit under strict and flexible assignments for two day sets. The flexible assignments shortens travel times per patient above 10% most of the times compared to the strict assignments. All differences among results are statistically significant.

Table 3.23: Travel times per visit under strict and flexible assignments for two day sets

| | | Day Set 1 | | | Day Set 2 | | |
|---|---|---|---|---|---|---|---|
| Region | Interarrival times | Strict | Flexible | % | Strict | Flexible | % |
| Small | 510 | 15.46 | **13.80** | -10.7 | 17.36 | **14.99** | -13.6 |
| Small | 340 | 14.68 | **12.49** | -14.9 | 15.64 | **13.54** | -13.4 |
| Small | 255 | 13.68 | **11.34** | -17.1 | 13.92 | **12.39** | -11.0 |
| Large | 510 | 25.87 | **23.21** | -10.3 | 29.27 | **25.45** | -13.1 |
| Large | 340 | 24.42 | **22.02** | -9.8 | 25.34 | **23.22** | -8.4 |
| Large | 255 | 22.63 | **19.90** | -12.1 | 24.03 | **21.67** | -9.8 |

Although visiting patients at the same times during their service horizons is preferred by patients, violating this preference or constraint apparently increases average daily visits and decreases travel times per visit. Under consideration of high demand for HHC service and the number of rejected requests, flexible assignments can be an option for companies to service more patients without increasing their resources.

## 3.7 Patient Preference and Pricing Policy

In the experiments we carried out so far, patient visit days and times are decided by the algorithm. The aim is to find the best days and times combination to optimise the schedule by considering future requests. We assume that patients accept visit days and times that we provide. However, it is highly possible that patients want to select visit days and times according to their schedules. Of course, their preferences are most likely not the best when we attempt to optimise the route and schedule. In this case, HHC providers choose to accept preferences of patients if patients are willing to pay more. The question then arises, how can the cost of a patient's visit times preference be calculated? If we had this cost, then we could use it in pricing of the service in real time. In this section, we estimate the cost to a provider of allowing patients to select day/time.

To be able to calculate the cost, we have to estimate how many visits we lose if we assign visits according to the preferences instead of visits that our algorithm provides. We simply run two simulations, corresponding to either scheduling according to preferences of patients or based on the company's assignments. It is important that other parameters such as randomly generated requests, other patients' arrival times, locations, weekly visits frequencies, and etc. are identical in both simulations so as to be able to compare them. We compare both simulation results in terms of total visits during the service horizon of the patient and charge the customer according to the difference. For example, if we make three visits less under patient preference days and times, we charge the patient considering those three visits. Calculation steps are as the following:

- Put times and days that a patient prefers into the existing schedule, if possible, during the service horizon,
- Simulate during the service horizon under dynamic patient demand,
- Count the number of visits and travel times,
- Let the algorithm assign that patient's visits during the service horizon,
- Simulate during the service horizon with patient arrivals that we used in the previous simulation,
- Count the number of visits and travel times.

In the test, the service area is large and the nurse is located at the centre of the area. We have a patient that is located at (14,9) and needs 3 visits each week during his or her service horizon. Two different scenarios are defined according to preferences of the patient. In the first scenario, the patient selects Tuesday, Wednesday, and Friday and 15.30, 10.30, and 15.15, days and times respectively. In the second scenario, selected days and times are Monday, Tuesday, Thursday and 11.00, 12.45, 13.00. In the visit days and times that the algorithm provides, the patient is assigned to Monday, Tuesday, and Wednesday, 10.45, 10.45, and 10.15. We assign two randomly generated visits to each day during the service horizon. 4-week and 8-week service horizons are assigned to requests.

Table 3.24: Total number of visits in both scheduling methods during 4-week and 8-week service horizons for scenario 1

| | **4 weeks** | | | **8 weeks** | | |
|---|---|---|---|---|---|---|
| **Interarrival times** | **Preference** | **SBA** | **Difference** | **Preference** | **SBA** | **Difference** |
| **510** | 141.77 | 144.10 | **2.33** | 310.10 | 313.17 | **3.07** |
| **340** | 154.13 | 156.27 | **2.13** | 318.07 | 322.67 | **4.60** |
| **255** | 160.27 | 164.13 | **3.87** | 330.13 | 340.47 | **9.33** |

Table 3.25: Total number of visits in both scheduling methods during 4-week and 8-week service horizons for scenario 2

| | **4 weeks** | | | **8 weeks** | | |
|---|---|---|---|---|---|---|
| **Interarrival times** | **Preference** | **SBA** | **Difference** | **Preference** | **SBA** | **Difference** |
| **510** | 141.77 | 144.10 | **2.33** | 311.93 | 313.17 | **1.23** |
| **340** | 155.20 | 156.27 | **1.07** | 318.83 | 322.67 | **3.83** |
| **255** | 162.07 | 164.13 | **2.07** | 335.43 | 340.47 | **5.03** |

As expected, assignments of SBA always allow more total visits in both scenarios as seen in Table 3.24 and 3.25. When the demand and service horizon increase, the gap between total visits increases as well. Under longer service horizons and higher demands, we expect that preferences of patients negatively affect total visits. However, different preferences can influence total visits significantly. For example, the preference in Scenario 1 causes 5 fewer visits under high demand and 8-week service length compared to Scenario 2. The suggested strategy here is to charge patients based on the number of less visits they cause. For example, if the patient needs 8-week service and the demand is high, we should charge the patient for regular visit cost and around extra 10 visits that we lost due to the patient visit days and times selection. Note that we only focus on total visits by ignoring travel times since our objective is to maximise patient visits.

In the above example, we consider how only one patient preference affects the schedule during his service horizon. Each preference can change the total daily visits and travel times dramatically as can be seen results of Scenario 1 and 2. However, there are also many factors to affect daily visits and travel times. For example, the number of weekly visits of a patient can be an important factor since days/times preferences of a patient who needs to be visited three times in a week has more effect on the schedule during his or her service horizon than the preference of a patient who needs only one visit per week. The other factor is the workload of a nurse. If the schedule of the nurse is totally empty or there are few visits in it, assigning visits of the patient based on his or her preference unlikely have a big effect on acceptance decisions and visits of future patients since there are most likely large gaps between its predecessors and successors that can be used to assign future patients' visits.

By considering above factors, it is better to test one year simulation horizon instead of testing scenarios based on each factor. We apply the procedure that calculates the difference between total visits of assignment of SBA and assignment based on a patient preference for all patients during the simulation horizon. Whenever a new patient arrives, he or she randomly chooses visits' days and times among all available days and times in the current schedule. Next, the algorithm starts a secondary simulation lasting the service horizon of the patient after we have taken the current schedule of the nurse and the patient's information such as the number of weekly visits, location, service horizon, and selected visits days/times from the main simulation. We calculate the total visits we have after the simulation has finished. Next, we let the algorithm assign the patient's visits and run the same simulation with the same parameters again. Finally, we calculate the total number of visits and the difference between visits of assignment of SBA and assignment based on the preference. The secondary simulation returns that difference to the main simulation. After that, the patient is assigned to the schedule according to his or her preference. We repeat this procedure for each arriving patient and accumulate differences during the simulation horizon. Note that the patient can select days/times from only available days/times in the schedule.

Accumulated differences represent visits we have lost due to preferences of patients. Therefore, we expect that summing up the total number of visits based on preferences of patients during the year and the total differences should more or less equal to the total number of visits in which all assignment decisions are made based on SBA.

Table 3.26 demonstrates a year period results according to the preference based and assignments of SBA. "Extra Visits" represents summing up visits we have lost due to preferences of patients during a year. Results in Table 3.24 and 3.25 come from replications running during only one patient's service horizon. However, we apply this procedure for hundreds of patients arrived during a year in this setting. "Total Visits" is the total number of visits for a year based preference based assignments and "Extra Visits". As explained above, we expect that "Total Visits" should be more or less equal to the total number of visits based on assignments of SBA. Note that we use the same simulation setting as in Table 3.3. All experiments in this section are carried out by using day set 1 since we assume that patients can select any days in a week.

It turns out that differences between results are mainly not statistically significant under different demands and area sizes. Costs of patient preference based

Table 3.26: Comparison of the preference based assignments with the assignments of SBA in terms of total number of visits

| Area size | Interarrival times | Preference | Extra visits | Total visits | SBA |
|---|---|---|---|---|---|
| Small | **510** | 2395 | 373 | 2768 | 2656 |
| | **340** | 2638 | 374 | 3012 | 3005 |
| | **255** | 2736 | 384 | 3120 | 3132 |
| Large | **510** | 2076 | 304 | 2381 | 2314 |
| | **340** | 2250 | 315 | 2566 | 2510 |
| | **255** | 2323 | 342 | 2665 | 2637 |

scheduling are roughly 15% less total number of visits for a year. We can also conclude that manual patient assignments performed by a nurse provide more or less the same results as preference based assignments since manual assignments mostly concern feasible scheduling more than optimisation. Table 3.27 shows average daily visits, travel times per visit, and acceptance rates according to preference based and assignments of SBA based on day set 1 and the small service area. Preference based assignments cause more than one visit lost, around 16% longer travel time per visit, and 6% less patient acceptance compared to assignments of SBA.

Table 3.27: Average daily visits, travel times per visit, and acceptance rates according to preference based assignments and assignments of SBA

| | **Average daily visits** | | **Travel times per visit** | | **Acceptance rates** | |
|---|---|---|---|---|---|---|
| **Interarrival times** | **Preference** | **SBA** | **Preference** | **SBA** | **Preference** | **SBA** |
| **510** | 7.48 | 8.30 | 18.36 | 15.54 | 0.78 | 0.84 |
| **340** | 8.24 | 9.39 | 17.61 | 14.25 | 0.59 | 0.65 |
| **255** | 8.55 | 9.79 | 16.84 | 13.09 | 0.46 | 0.52 |

We proposed an algorithm based on SBA in order to price a patient's preference of visit times and days. The main idea was to calculate the difference between the number of visits based on times and days that a patient selects and times and days assigned by SBA. We tested the idea under different service horizons and interarrival times. Finally, we tested the idea for one year simulation horizon to be able to see whether or not overall it works under all situations we can encounter during a planning horizon. In practice, HHC companies can provide different prices for each patient based on how many visits are lost due to days and times patients select or a standard average price based on how many visits a patient needs and an average cost per visit. For example, according to assignments based on patients' preferences in Table 3.26, a nurse performed 2395 visits in the small area and under the low demand during a year. The company lost 373 visits due to visit days and times patients preferred. The average cost per visit is 0.16 (373/2395). Let us assume that the company has a patient request who needs three weekly visits and 8 week service horizon. So he or she needs to be serviced 24 times overall. The extra charge should be value of 3.73 (0.16*24) visits. Our algorithm supports both pricing policies. Note that we ignore travel time cost and rejection possibility of a patient to feasible times and days we provide.

# Chapter 4

# HHC Model for Multiple Nurses

In the previous chapter, SBA has been demonstrated for a single nurse who is travelling and servicing in a specified area. In this section, more realistically, we apply SBA to the case where there is more than one nurse. Several methodologies are applied to find optimum solutions in terms of total daily visits of all nurses and results are compared to the distance heuristic. After explaining solution methodologies, we explain why the solution methodology for a single nurse does not work well for multiple nurses by simply dividing the service area and total demand according to the number of nurses. Next, we examine how different visit durations and violation of service continuity affect results. Finally, we demonstrate a simple pricing policy based on patient preferred visit days, times, and nurses at the end of this chapter.

## 4.1 Distance and Capacity Heuristics for Multiple Nurses

The distance heuristic for multiple nurses (DHM) is similar to the distance heuristic for the single nurse (DH). We make small modifications to work with more than one nurse. Whenever a new patient arrives to the system, the algorithm calculates the cost of insertion of that patient between all requests assigned already in each day of the week and nurse. After that, the method finds intervals with the cheapest insertion costs according to visit frequency of the patient for each nurse and sum them up to be able to select the nurse with the smallest insertion cost. Finally, all visits are scheduled to those cheapest days and time slots of the nurse during the service horizon of the patient. The algorithm assigns visits of patients to time slots right before or after their successors or predecessors in terms of their proximity to them. If there are several nurses which have the same insertion costs, as a tie-breaker, we assign the visit to one where fewer patient visits are already scheduled to balance the workload of nurses.

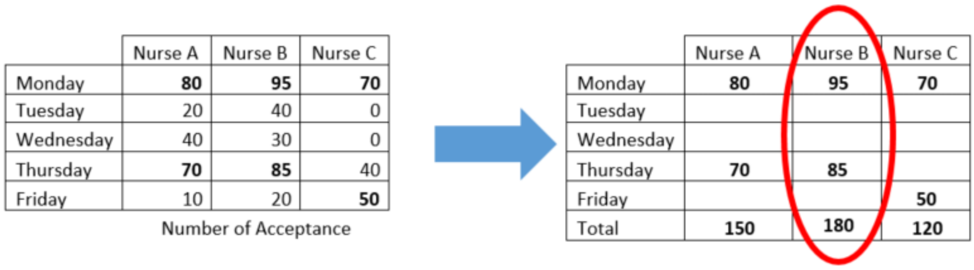
Results in both [Bennett and Erera, 2011] and the previous chapter showed that the distance heuristic outperformed the capacity heuristic under high demand, large uniform, and large uniform-clustered areas for the single nurse case. In the problem settings of this study, we test our algorithm only in the large area and high demand case since we have many nurses. Therefore, we only compare our algorithm with the distance heuristic.

## 4.2 Extended SBA

We use SBA for the multiple nurse case in the same way as for the single nurse case. According to the approach, we generate several scenarios for each nurse independent of other nurses. This procedure is applied for each day in the week. As a result, we find how many times and which time slots a request is assigned for each nurse and day. To select the most suitable nurse for a request, we simply compare the number of acceptances. The nurse who has the highest number of acceptances over all scenarios and weekly visit days are assigned to the patient. If some nurses have an equal number of acceptances, distances between nurses and the request are used for tie-breaking. In this condition, the nearest nurse to the request is selected. The following example illustrates the above proposal. Let's assume that there are three nurses, A, B, and C located at different parts of the service area. Moreover, assume that a request arrives on Monday from a random location in the service area and demands two visits per week. Episode of care and service duration are not considered since they are same for all patients. Now we have to decide whether we accept or reject the request. Firstly, for nurse A, we generate several scenarios for each day of the next week. Each scenario has a number of randomly generated requests and the current request. To find how many requests we need to generate randomly, we calculate the average weekly demand. If we are looking at next Monday and the expected weekly demand is 12 new patient requests, the total number of visits for next week equals 30 (12*2.5), where 2.5 is the expected visit frequency per week for a patient. Thus, we divide the total number of weekly visits by 5 to find the average number of visits for a day and by 3 to find the demand per nurse. It means that 2 requests are generated for each scenario and the current request is added to them. After that, we try to construct a tour by considering the requests in the scenario and those previously assigned for that day and nurse. Requests are being assigned to the tour by the cheapest insertion heuristic until the tour is full or all requests in the scenario have been scheduled. Finally, we check whether the current request has been scheduled and if so, in which time slot he or she has been scheduled. After all scenarios have been simulated, we find how many times the new request is accepted and which time slot he or she has been frequently assigned for that day. To decide which day combination (Monday-Friday, Tuesday-Thursday, etc.) he or she is scheduled, we pick up the best one, two or three days in terms of the number of assignments over all scenarios. Next, we repeat the same process for nurse B and C, and suppose that the number of acceptances for the best day combinations are 150, 180, and 120, for nurse A, B, and C respectively. We choose nurse B since he or she has the highest number of acceptances for the request. As can be seen in Figure 4.1, the best nurse and service days for the request are time slots of nurse B on Monday and Thursday. If there is a nurse to whom the request is never assigned over all scenarios, the request is ignored during the comparison. Of course, if the request cannot be assigned to any nurse in any scenario, the request is rejected. Under condition of equal acceptances for several nurses, distances between the request and nurses are used for tie-breaking. The patient is assigned to the nearest nurse.

### 4.2.1 Simulation Settings

Simulation settings are similar to the settings for the single nurse case. 30 replications are run for each experiment. Each replication lasts 360 days where each day takes 510 minutes. A day is composed of 35 time slots. Duration between two time slots is 15 minutes. Interarrival times between requests are exponentially distributed with mean 510, 340, 255, and 150 minutes. We add a four-week warm up period at the beginning of each experiment. The patient requests to be serviced with stochastic visit frequency 1, 2, or 3 visits per week with probabilities 0.05, 0.35, and 0.60, respectively. The service horizon lasts 4 weeks for each accepted patient. However, we will change the duration for later trials which is explained in the related sections. We have two different set-ups for visit days each patient can be assigned to according to his or her weekly visit frequency. At the first set-up, day set 1, each patient can be scheduled any combination of days in the week. Day combinations for a patient can be $\binom{n}{f}$ when $f$ represents the visit frequency and $n$ shows the number of days (Monday, Tuesday, Wednesday, Thursday, Friday). On the other hand, we also use another day set-up, day set 2, which does not allow to schedule sequential days when the visit frequency of a patient is two or three since it is not realistic to perform some tasks such as cooking, bathing, etc. the first two or three days at the beginning of a week and to do nothing at the remaining days. Therefore, only following visit day combinations can be assigned to a patient who needs two visits in a week, *((Monday,Friday),(Monday,Thursday),(Monday,Wednesday), (Tuesday,Friday),(Tuesday,Thursday))* and a patient who needs three visits in a week, *(Monday-Wednesday-Friday)*. First visit starts the following week after the request is accepted. Visit durations are deterministic and take 30 minutes. Each arriving customer request is equally likely to arise in a square geographic region subdivided into 3600 equally-sized square subregions. The nurses are located at (10,10), (30,30), and (40,50) for three-nurse case and (10,10), (30,30), (40,50), (60,30), (20,20), and (3,55) for six-nurse case in the region. To understand differences between results, we construct independent samples t-test and calculate p-values for each pair. Numbers



| | Nurse A | Nurse B | Nurse C |
|---|---|---|---|
| Monday | **80** | **95** | **70** |
| Tuesday | 20 | 40 | 0 |
| Wednesday | 40 | 30 | 0 |
| Thursday | **70** | **85** | 40 |
| Friday | 10 | 20 | **50** |

Number of Acceptance

| | Nurse A | Nurse B | Nurse C |
|---|---|---|---|
| Monday | **80** | **95** | **70** |
| Tuesday | | | |
| Wednesday | | | |
| Thursday | **70** | **85** | |
| Friday | | | **50** |
| Total | **150** | **180** | **120** |

Figure 4.1: Nurse and day selection process for multi-nurse case

written in bold font mean that they are statistically better.

### 4.2.2 Results

Table 4.1 and 4.2 represent total daily visits and travel times per visit for three nurses according to SBA, DHM and different visit day sets. Statistically there is no significant difference between results when interarrival times are 510 and 340 minutes, but SBA works better when the interarrival time is 255 minutes for day set 2 and 150 minutes for day set 1. In terms of travel times per visit, SBA provides significantly lower travel times compared to DHM's travel times. Furthermore, SBA distributes patient visits to three nurses mostly equally while distributions of daily visits seem unfair in terms of workload balance of nurses in DHM. It is highly possible to assign a request to a nurse whose tour is busier than others since the chance to find the cheapest insertion in a busy tour is higher in DHM. However, we generate random requests for each nurse when constructing tours during the scenario phase in SBA. Therefore, the chance of assigning a request to a busy nurse decreases since most likely more suitable random requests are scheduled to the remaining nurses. We find a visit range for each run, which is the difference between maximum and minimum daily visits of nurses. For example, if Nurse 1, 2, and 3 visit on average 4, 5, and 6 patients in a run, the range for this trial is 2 visits. After that, we calculate the average range for 30 runs and a confidence interval for the mean. Means closer to zero are more desirable since they indicate that average daily visits of nurses are

Table 4.1: Total daily visits and travel times per visit for SBA and DHM when employing three nurses using day set 2

| Interarrival times | Method | Range | Total daily visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|
| 510 | SBA | 0.73 ±0.09 | 10.38 | 0 | **26.23** | -11.29 |
| | DHM | 1.60 ±0.09 | 10.40 | | 29.57 | |
| 340 | SBA | 0.51 ±0.06 | 14.42 | 0 | **25.09** | -13.80 |
| | DHM | 1.83 ±0.08 | 14.36 | | 29.10 | |
| 255 | SBA | 0.35 ±0.07 | **17.05** | 4.18 | **23.20** | -20.77 |
| | DHM | 2.03 ±0.07 | 16.36 | | 29.28 | |
| 150 | SBA | 0.28 ±0.05 | 19.95 | 0 | **19.21** | -33.84 |
| | DHM | 1.41 ±0.12 | 20.06 | | 29.03 | |

Table 4.2: Total daily visits and travel times per visit for SBA and DHM when employing three nurses using day set 1

| Interarrival times | Method | Range | Total daily visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|
| 510 | SBA | 1.79 ±0.11 | 10.49 | 0 | **24.70** | -15.25 |
| | DHM | 1.92 ±0.20 | 10.45 | | 29.14 | |
| 340 | SBA | 0.43 ±0.08 | 15.63 | 0 | **23.02** | -13.21 |
| | DHM | 1.03 ±0.13 | 15.88 | | 26.52 | |
| 255 | SBA | 0.27 ±0.05 | 20.01 | 0 | **18.97** | -26.50 |
| | DHM | 0.38 ±0.06 | 19.98 | | 25.81 | |
| 150 | SBA | 0.23 ±0.03 | **24.32** | 2.79 | **16.93** | -33.23 |
| | DHM | 0.17 ±0.04 | 23.66 | | 25.36 | |

almost equal. ”Range” in tables shows average ranges and confidence intervals with 95% confidence level. The range exceeds 2 visits per day in DHM when interarrival time is 255 minute. In real life, it is not tolerated that one nurse in the company visits 2 patients more than another nurse every day if they are paid the same wage.

Table 4.3 and 4.4 show total daily visits and travel times per visit for six nurses according to SBA and DHM. Total daily visits of six nurses are not statistically different for both methods. The most important reason is that the number of nurses is adequate to accept all patient requests. However, travel times per visit in SBA sharply decrease compared to DHM travel times. When the interarrival time is 150 minutes, SBA decreases travel time almost 40% compared to DHM for day set 1 and around 32% for day set 2. All patients are accepted under lower demands (510 and 340) while acceptance rate goes down to 91% when the demand is high for day set 2.

Although SBA is at least as good as DHM in terms of average daily visits and much better in terms of travel times per visit under a variety of scenarios, there is an important drawback of SBA for the multiple-nurse case. How to distribute an existing demand to nurses is a great problem for this method. Previous tests, we simply distributed the demand to nurses equally. For example, If we expect 12 new patients (30 weekly visits) for the next week, expected visits for each day are 6 and for each nurse are 2 if we have 3 nurses. However, based on the current tours of nurses, it is highly possible that more or less than 2 visits can be assigned to one of nurses. Moreover, when the demand is low, most likely, the number of expected visits for each nurse becomes less than 1 visit for each day. In this situation, we have to round down or up zero or one and this can affect results significantly. Another method can be to generate the number of random request for each nurse based on the total expected daily visits. By considering the above example, we generate 6 visits for each nurse instead of 2. At this time, when the demand is high, we generate many random requests and the probability that a patient is accepted dramatically decreases. Therefore, we modify SBA to be able to consider all nurses and randomly

Table 4.3: Total daily visits and travel times per visit for SBA and DHM when employing six nurses using day set 1

| Interarrival times | Method | Total daily visits | Travel time per visit | % | Acceptance rate (%) |
|---|---|---|---|---|---|
| 510 | SBA | 10.50 | **20.24** | -28.9 | 100 |
| | DHM | 10.45 | 28.46 | | 100 |
| 340 | SBA | 15.72 | **18.30** | -30.2 | 100 |
| | DHM | 15.91 | 26.23 | | 100 |
| 255 | SBA | 21.00 | **18.15** | -27.5 | 99.60 |
| | DHM | 20.94 | 25.03 | | 100 |
| 150 | SBA | 34.55 | **14.19** | -39.5 | 99.86 |
| | DHM | 35.50 | 23.45 | | 99.97 |

Table 4.4: Total daily visits and travel times per visit for SBA and DH when employing six nurses using day set 2

| Interarrival times | Method | Total daily visits | Travel time per visit | % | Acceptance rate (%) |
|---|---|---|---|---|---|
| 510 | SBA | 10.50 | **20.07** | -27.5 | 100 |
| | DHM | 10.52 | 27.67 | | 100 |
| 340 | SBA | 15.73 | **19.93** | -23.4 | 100 |
| | DHM | 15.87 | 26.03 | | 100 |
| 255 | SBA | 21.09 | **20.13** | -19.9 | 99.73 |
| | DHM | 20.92 | 25.12 | | 99.98 |
| 150 | SBA | 32.39 | **18.23** | -31.5 | 92.53 |
| | DHM | 32.23 | 26.67 | | 91.93 |

generated requests together when generating scenarios and making decisions at the next section.

## 4.3 Scenario Based Approach for Multiple Nurses (SBAM)

As it is explained in the previous section, evaluating each nurse independently is an oversimplification even though overall results are at least as good as results of the greedy algorithm. In this section, a new approach that considers all nurses and randomly generated requests at the same time when generating scenarios is proposed. The basic idea behind the approach is when constructing tours for each scenario, assignment of random requests and actual one are done by considering the existing tours of each nurse. Now we explain more in detail.

First, we again start to generate several scenarios for the first day of the week. Cost of assigning the request to each time slot of each nurse is calculated by using the cheapest insertion heuristic. After calculating assignment cost of each random request and the current request, the one who has the lowest insertion cost is selected and assigned to the schedule. This scheduling lasts until all randomly generated requests and the actual one have been assigned or no further request can be inserted into any tour. Next, whether the actual request has been assigned and which time slot it has been scheduled in is recorded. After repeating this procedure for a pre-defined number of scenarios, we determine how many times the request has been assigned to each nurse and which time slots it has been scheduled in. We reiterate the same process and find the number of acceptances and visit times for the remaining days of the week. Finally, comparisons between nurses are performed in terms of the total number of acceptances in terms of the patient's weekly visits. If the request cannot be assigned to any nurse, it is rejected. The following example shall demonstrate the process.

Let us again assume that there are three nurses, A, B, and C, who live in different parts of the service area and their Monday schedules are presented in Figure 4.2. Moreover, assume that a patient request arrives on Monday from a random location

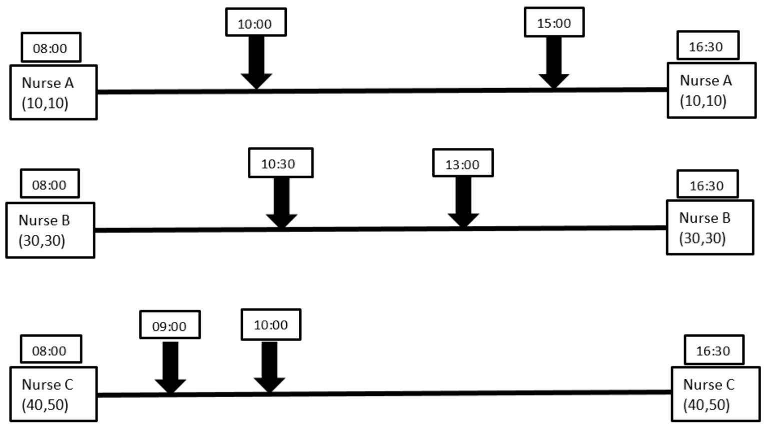



Figure 4.2: Monday schedules of Nurse A, B, and C

in the service area with 2-day-visit frequency. First, we have to calculate the demand for Monday. If the demand is predicted 6 patients for the next week, the total visits for the week will be 15 (6*2.5 (the expected weekly visits for each request) ) and 3 visits (15/5) will be expected for each day. It means that 3 requests are generated for each scenario on Monday and we add our actual request, of course.

Next step is to calculate insertion cost of each request. Insertion costs of each request in the scenario to feasible time slots of nurses are calculated by the cheapest insertion heuristic as explained in Section 3.2.1. SBAM calculates the cost of each insertion to each feasible time slot of nurses as demonstrated in Figure 4.3. $C_{A1}$ represents insertion cost of a request to the first time interval of nurse A. After calculating the cost for each feasible time interval of nurse A, the algorithm picks up the cheapest one, $C_{A2}$. Next, the same calculations are done for nurse B and C, and the cheapest insertions are found. In our example, let us assume that the cost order is $C_{C1} < C_{B3} < C_{A2}$. So $C_{C1}$ is the cheapest cost for the request. However, we have 3 randomly generated and the actual request in the scenario. Therefore, we need to calculate and find the cheapest cost for each. After finding the request that has the cheapest insertion cost among all requests in the scenario, the algorithm removes it from the scenario and adds it into the time interval of the nurse observed

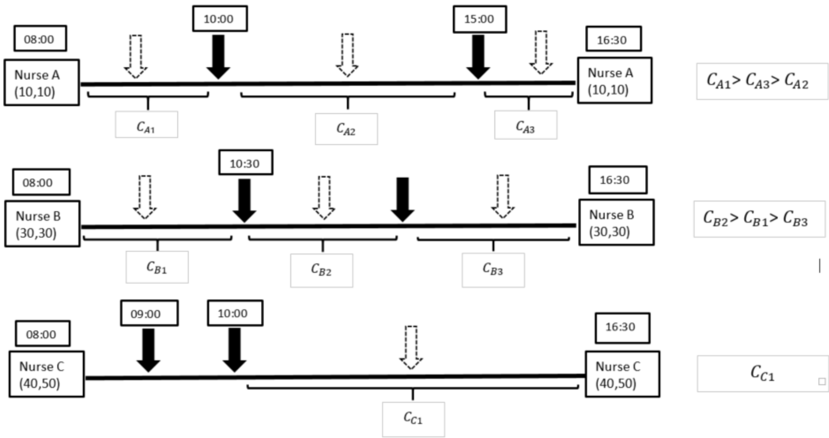



Figure 4.3: Nurse and day selection by finding the interval with the cheapest insertion cost

before. This procedure is repeated until all requests in the scenario are assigned or no feasible time interval exists.

This procedure is repeated for a given number of scenarios. After that, the algorithm produces a record as in Table 4.5. So we know how many times, when, and which nurses the request has been assigned to for only Monday. The same information is produced for the remaining days to be able to compare and decide the most suitable nurse and day combination. The results may look similar to Figure 4.1. In our example, the patient needs 2 visits in a week. The best choice is that the patient should be served by nurse B on Mondays and Thursdays during his or her service horizon since the total number of acceptances is the highest (100 scenarios are generated for each day in the example.).

Table 4.5: Assignments times for each nurses

| Scenario No | 1 | 2 | 3 | 4 | 5 | .... | n |
|---|---|---|---|---|---|---|---|
| Assignment | Yes | Yes | No | Yes | No | .... | Yes |
| Nurse | C | A | .... | C | .... | .... | B |
| Time | 10:45 | 11:00 | .... | 10:45 | .... | .... | 12:00 |

Table 4.6: Total daily visits and travel times per visit for SBAM, SBA and DHM, and the percentage changes between daily visits and travel times of SBAM and SBA when employing three nurses for day set 1

| Interarrival times | Method | Range | Total daily visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|
| 510 | SBA | 1.79 ±0.11 | 10.49 | 0 | **24.70** | -12.17 |
| | SBAM | 1.70 ±0.27 | 10.49 | | **21.69** | |
| | DHM | 1.92 ±0.20 | 10.45 | | 29.14 | |
| 340 | SBA | 0.43 ±0.08 | 15.63 | 0 | **23.02** | -12.48 |
| | SBAM | 1.06 ±0.15 | 15.69 | | **20.14** | |
| | DHM | 1.03 ±0.14 | 15.88 | | 26.52 | |
| 255 | SBA | 0.27 ±0.05 | 20.01 | 2.79 | **18.97** | 2.67 |
| | SBAM | 0.36 ±0.08 | **20.57** | | **19.48** | |
| | DHM | 0.38 ±0.06 | 19.98 | | 25.81 | |
| 150 | SBA | 0.23 ±0.03 | 24.32 | 6.68 | **16.93** | 5.31 |
| | SBAM | 0.19 ±0.03 | **25.95** | | **17.83** | |
| | DHM | 0.17 ±0.04 | 23.66 | | 25.36 | |

Table 4.6 represents total daily visits and travel times per visit for the three nurse case according to SBAM, SBA, and DHM for day set 1. For 510 and 340 minutes interarrival times, SBAM gives statistically the same average daily visits, but shorter travel times than DHM and SBA. SBAM decreases travel times by 12% and 25% compared to SBA and DHM respectively. In higher demand cases (255 and 150 minutes), increases of total daily visits are remarkable than DHM and SBA. Although SBAM ensures more than one additional visit than SBA and two additional visits than DHM daily when the interarrival time is 150 minutes, travel times per visit are slightly longer than SBA but much shorter than DHM. It is important to remember that an increase in daily visits by one means 360 extra visits in a year.

Table 4.7 shows total daily visits and travel times per visit for a six-nurses case according to SBAM, SBA and DHM for day set 1. Again total daily visits are not different for all methods in case of low demands (510 and 340 minutes). However, travel times per visit of SBAM are much shorter than travel times provided by SBA and DHM. When the demand is getting higher (255 and 150 minutes), differences between average daily visits are becoming significant as well. Furthermore, SBAM still gives the shortest travel times in the last two cases. It is worthwhile to emphasize that acceptance rates in Table 4.7 are around 100% which mean that almost all patient requests are accepted. One may wonder why smaller interarrival times are not tested. An interarrival time with mean 150 minutes equals 3.4 requests per day from a region of 3600 square kilometres. This is a quite reasonable demand when compared with demands in some application studies [Bennett and Erera, 2011] [Duque et al., 2015]. If the demand overcomes service availability significantly, of course, HHC providers will employ more nurses to be able to fulfil them as soon as possible.

Table 4.8 and 4.9 represent total daily visits and travel times per visit for the three and six-nurses cases according to SBAM, SBA and DHM for day set 2. Different from day set 1 cases, SBAM performs worse compared to DHM and SBA especially in low demand scenarios. It is only competitive when the interarrival time is 150 minutes. It can happen that the new patient is allocated a visit by one nurse on one

Table 4.7: Total daily visits, travel times per visit, and acceptance rates for SBAM, SBA and DHM, and the percentage changes between travel times of SBAM and SBA when employing six nurses for day set 1

| Interarrival times | Method | Total daily visits | Travel time per visit | % | Acceptance rates (%) |
|---|---|---|---|---|---|
| 510 | SBA | 10.50 | **20.24** | -11.67 | 100 |
| | SBAM | 10.49 | **17.88** | | 100 |
| | DHM | 10.45 | 28.46 | | 100 |
| 340 | SBA | 15.72 | **18.30** | -12.84 | 100 |
| | SBAM | 15.70 | **15.95** | | 100 |
| | DHM | 15.91 | 26.23 | | 100 |
| 255 | SBA | 21.00 | **18.15** | -19.11 | 100 |
| | SBAM | **21.61** | **14.68** | | 100 |
| | DHM | 20.94 | 25.03 | | 100 |
| 150 | SBA | 34.55 | **14.19** | -7.68 | 99.86 |
| | SBAM | 35.74 | **13.10** | | 99.98 |
| | DHM | 35.50 | 23.45 | | 99.97 |

Table 4.8: Total daily visits and travel times per visit for SBAM, SBA and DHM, and the percentage changes between daily visits and travel times of SBAM and SBA when employing three nurses for day set 2

| Interarrival times | Method | Range | Total daily visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|
| 510 | SBA | 0.73 ±0.09 | 10.38 | 0 | **26.23** | -10.23 |
| | SBAM | 0.82 ±0.17 | 9.95 | | **23.55** | |
| | DHM | 1.60 ±0.09 | 10.40 | | 29.57 | |
| 340 | SBA | 0.51 ±0.06 | 14.42 | 0 | **25.09** | -6.87 |
| | SBAM | 0.74 ±0.13 | 13.99 | | **23.36** | |
| | DHM | 1.83 ±0.08 | 14.36 | | 29.10 | |
| 255 | SBA | 0.35 ±0.07 | **17.05** | -0.96 | **23.20** | -1.82 |
| | SBAM | 0.59 ±0.10 | 16.88 | | **22.77** | |
| | DHM | 2.03 ±0.07 | 16.36 | | 29.28 | |
| 150 | SBA | 0.28 ±0.05 | 19.95 | 6.88 | **19.21** | 5.81 |
| | SBAM | 0.22 ±0.05 | **21.32** | | **20.32** | |
| | DHM | 1.41 ±0.12 | 20.06 | | 29.03 | |

Table 4.9: Total daily visits, travel times per visit, and acceptance rates for SBAM, SBA and DHM, and the percentage changes between travel times of SBAM and SBA when employing six nurses for day set 2

| Interarrival times | Method | Total daily visits | Travel time per visit | % | Acceptance rates (%) |
|---|---|---|---|---|---|
| 510 | SBA | 10.50 | **20.07** | -10.12 | 100 |
| | SBAM | 10.32 | **18.04** | | 98.57 |
| | DHM | 10.52 | 27.67 | | 100 |
| 340 | SBA | 15.73 | **19.93** | -16.41 | 100 |
| | SBAM | 14.58 | **16.66** | | 98.18 |
| | DHM | 15.87 | 26.03 | | 100 |
| 255 | SBA | **21.09** | **20.13** | -22.16 | 99.73 |
| | SBAM | 20.51 | **15.67** | | 97.61 |
| | DHM | 20.92 | 25.12 | | 99.98 |
| 150 | SBA | 32.39 | **18.23** | -8.01 | 92.53 |
| | SBAM | 32.32 | **16.77** | | 89.93 |
| | DHM | 32.23 | 26.67 | | 91.93 |

day, and another nurse on another day, which violates the consistency constraint. Let us assume that we have three nurses, A, B, and C, and a new request that needs three day visits. We create 75 scenarios for each day in the week and the number of acceptances for each nurse can be seen in Table 4.10.

Table 4.10: The number of acceptance for each nurse

| | Monday | Tuesday | Wednesday | Thursday | Friday |
|---|---|---|---|---|---|
| Nurse A | 75 | 0 | 0 | 0 | 0 |
| Nurse B | 0 | 0 | 75 | 0 | 0 |
| Nurse C | 0 | 0 | 0 | 0 | 75 |

In this case, although we have many available slots, we have to reject the patient since none of the nurses seems available for Monday, Wednesday, and Friday. According to our experiments, the possibility to encounter tables as above highly increases under low demands. Therefore, we also need to look at a model where future requests were generated with weekly visits, and inserted into the week as one combination.

## 4.4 Modification of SBAM

Because of the drawback of SBAM as explained above, it is necessary to modify it to be able to improve its performance. The idea behind SBAM is to consider all nurses and randomly generated requests at the same time when generating scenarios. However, we consider each day of the week separately, independent of several visits of each patient in that week. When more nurses start to be taken into consideration, it causes a problem that one visit of a patient is assigned to one nurse while other visits can be assigned to another nurse. As it is emphasized from the beginning, service consistency, i.e. a patient is visited same days and times by the same nurse during his or her service horizon, is an important constraint. Therefore, we have to consider all patient visits in the week simultaneously when generating scenarios to be able to eliminate this drawback and keep service continuity. The method we will use is similar to WSBA in Section 3.3.2. The modification is to consider all nurses instead of a single nurse when calculating cost risen from insertion of all weekly visits of requests. In other words, to find which nurse's tours are the most suitable for all weekly visits of a request, the algorithm looks at the smallest cost over all possible insertions into each nurse's routes and calculates the total insertion cost for all weekly visits and the average insertion cost per visit. Algorithm 4.1 shows the pseudo code for SBAM. "nCombinations" represents the days for which visits of a patient can be scheduled and "nReqInTour" represents how many times the request has been assigned over all scenarios. Note that we use the same number of scenarios and acceptance threshold as we did for the single nurse case.

Let's give a concrete example to make the method more understandable. We have three nurses, A, B, and C, and a request R that needs three visits per week. We generate scenarios to decide whether or not we accept request R. The algorithm generates 5 random requests according to expected weekly demand. Table 4.11 shows the assignment costs of random request 2 calculated with the cheapest insertion heuristic for each visit day and nurse. In this illustration, the request that needs three visits has to be scheduled on Monday, Wednesday, and Friday. After finding the total assignment cost by summing up daily costs for each nurse, we select the lowest one. In this example, Nurse A provides the cheapest insertion cost. We can observe the left side of Table 4.12 (Iteration 1) after calculating insertion cost for all requests in the scenario.

**Algorithm 4.1** Scenario Based Approach for Multiple Nurses **CIH:** Cheapest Insertion Heuristic, **M:** A large positive constant

```
TimeSlot ← ∅
nReqInTour ← 0
for i= 1 To n  do
    ScenarioSize ← WeeklyDemand
    Scenario + = CurrentRequest
    for j= 1 To ScenarioSize  do
```

```
    Scenario + = RandomlyGeneratedRequest
end for
Tour ← Existing Visits
while Tour is feasible and Scenario is not empty do
    MinGlobalCost ← ∞
    for k= 1 To ScenarioSize+1  do
        AverageCost ← 0
        MinNurseCost ← ∞
        for n= 1 To NumberofNurse  do
            MinDayCost ← ∞
            for p= 1 To nCombinations  do
                if CIH(Request[k],Nurse[n],p)=<MinDayCost then
                    MinDayCost←CIH(Request[k],Nurse[n],p)
                end if
            end for
            if MinNurseCost => MinDayCost then
                MinNurseCost←MinDayCost
            end if
        end for
        AverageCost ← MinNurseCost/Frequency[k]
        if AverageCost =< MinGlobalCost then
            MinGlobalCost←AverageCost
            index←k
            NurseIndex←n
        end if
    end for
    if Scenario[index] is feasible for Nurse[NurseIndex] Tour then
        Nurse[NurseIndex] Tour+ = Scenario[index]
        Remove Request from Scenario
```

```
        end if
    end while
    if CurrentRequest in anyTour then
        nReqInTour++
        TimeSlot ← CurrentRequestScheduleTime
    end if
end for
if nReqInTour>0 then
    Accept Patient
    Time ← MostFrequentTime (TimeSlot)
end if
```

Table 4.11: Assignment cost for each visit of random request 2 and total cost

| | Monday | Tuesday | Wednesday | Thursday | Friday | Total Cost |
|---|---|---|---|---|---|---|
| Nurse A | 50 | ... | 20 | ... | 20 | 90 |
| Nurse B | 30 | ... | 60 | ... | 50 | 140 |
| Nurse C | 70 | ... | 50 | ... | 70 | 190 |

"Visits" column shows how many weekly visits patients need. "Nurse" and "Total Cost" show the selected nurse at the previous step and the total insertion cost of all visits. "Average Cost" is calculated by dividing the total cost by the number of weekly visits. We need average cost to be able to compare visit cost of different requests to select the cheapest one. In the example, random request 2 has the cheapest insertion cost. Thus the algorithm chooses it to assign it to the weekly schedule with all its visits. It is removed from the scenario before iteration 2. Same calculation procedure is repeated in iteration 2 and results in Table 4.12 are observed. As can be seen, our actual request has the cheapest insertion cost this time and it is assigned to the weekly schedule with its three visits and removed from the scenario before iteration 3. Iterations last until all requests are scheduled or no more request can be inserted. After repeating this procedure for a predefined number of scenarios, we determine how many times the request has been assigned to each nurse and which time slots it has been scheduled in for each visit day. Finally, the algorithm selects the nurse/time slot combination.

Table 4.12: Total and average costs for each visit of all requests

| | Iteration 1 | | | | Iteration 2 | | |
|---|---|---|---|---|---|---|---|
| | Visits | Nurse | Total cost | Average Cost | Nurse | Total cost | Average Cost |
| RandomR1 | 1 | A | 50 | 50 | B | 60 | 60 |
| RandomR2 | 3 | A | 90 | **30** | | | |
| RandomR3 | 3 | B | 120 | 40 | B | 150 | 50 |
| RandomR4 | 2 | C | 100 | 50 | A | 120 | 60 |
| RandomR5 | 3 | C | 150 | 50 | C | 150 | 50 |
| RequestR | 3 | A | 105 | 35 | A | 100 | **33** |

Table 4.13 and 4.14 show total daily visits and travel times per visit for SBAM, SBA, and DHM when employing three and six nurses for day set 2 after the modification. First, when Table 4.13 and Table 4.8 that shows results before the modification are compared, it can be observed that total daily visits obviously increases under the same experimental setting. Moreover, comparison of Table 4.14 and Table 4.9 indicates that the modification successfully works for the six nurses. Thus, we can say that the modification improves results. We discuss further details about comparisons in the next section.

### 4.4.1 High Number of Nurses and Longer Service Horizon/Time

We tested the algorithms for 3 and 6 nurses so far. Note that as we increase the number of nurses, we have to increase demand proportionally in order to keep acceptance rates under 100%. Otherwise, we cannot understand whether or not our algorithm is superior since there are sufficient nurses to accept almost all requests for both methods. However, higher arrival rate increases computational cost substantially as shown in Figure 4.4. Instead, we increase visit times and the service

Table 4.13: Total daily visits and travel times per visit for SBAM, SBA and DHM, and the percentage changes between daily visits and travel times of SBAM and SBA when employing three nurses for day set 2 after the modification

| Interarrival times | Method | Range | Total daily visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|
| | SBA | 0.73 ±0.09 | 10.38 | | **26.23** | |
| 510 | SBAM | 0.48 ±0.09 | 10.46 | 0 | **23.91** | -8.83 |
| | DHM | 1.60 ±0.09 | 10.40 | | 29.57 | |
| | SBA | 0.51 ±0.06 | 14.42 | | **25.09** | |
| 340 | SBAM | 0.63 ±0.15 | 14.39 | 0 | **23.85** | -4.93 |
| | DHM | 1.83 ±0.08 | 14.36 | | 29.10 | |
| | SBA | 0.35 ±0.07 | 17.05 | | **23.20** | |
| 255 | SBAM | 0.59 ±0.08 | **17.36** | 1.82 | **22.80** | -1.71 |
| | DHM | 2.03 ±0.07 | 16.36 | | 29.28 | |
| | SBA | 0.28 ±0.05 | 19.95 | | **19.21** | |
| 150 | SBAM | 0.26 ±0.03 | **22.22** | 11.42 | **20.58** | 7.13 |
| | DHM | 1.41 ±0.12 | 20.06 | | 29.03 | |

Table 4.14: Total daily visits, travel times per visit, and acceptance rates for SBAM, SBA and DHM, and the percentage changes between travel times of SBAM and SBA when employing six nurses using day set 2 after the modification

| Interarrival times | Method | Total daily visit | Travel time per visit | % | Acceptance rates (%) |
|---|---|---|---|---|---|
| | SBA | 10.50 | **20.07** | | 100 |
| 510 | SBAM | 10.56 | **17.48** | -12.90 | 100 |
| | DHM | 10.52 | 27.67 | | 100 |
| | SBA | 15.73 | **19.93** | | 100 |
| 340 | SBAM | 15.81 | **16.14** | -19.02 | 100 |
| | DHM | 15.87 | 26.03 | | 100 |
| | SBA | 21.09 | **20.13** | | 99.73 |
| 255 | SBAM | 21.11 | **15.77** | -21.66 | 99.98 |
| | DHM | 20.92 | 25.12 | | 99.98 |
| | SBA | 32.39 | **18.23** | | 92.53 |
| 150 | SBAM | **33.41** | **17.23** | -5.49 | 94.31 |
| | DHM | 32.23 | 26.67 | | 91.93 |

horizon to 50 minutes and 8 weeks respectively. So caregivers spend 50 minutes for each visit and patients are served over 8 weeks instead of 4 weeks.

We test the SBAM for 12 and 24 nurses. In this trial, all nurses are homogeneous in terms of their qualifications. Location of nurses are assigned uniformly across the service area. Table 4.15 and 4.16 show total daily visits and travel times per visit for 12 and 24 nurses, respectively. For 12 nurses, there is no statistical difference between total daily visits of SBAM and DHM since the acceptance rate is 100% for both in the 255 minute interarrival time case. For other cases, SBAM is able to schedule significantly more visits than DHM, and the improvement is even larger for travel times per visit. All differences are statistically significant.

We have similar results for 24 nurses. Although the demand is more than 5 requests every day (interarrival time is 100 minutes), acceptance rates are 100%. When the acceptance rate is around 98%, SBAM provides 4 more visits than DHM every day. It is obvious based on previous experiments that the gap between total visits of SBAM and DHM is getting larger when the demand increases. On the other hand, travel times per visit shorten more than 50% with SBAM.

Now we can demonstrate several different extensions such as depot locations, clustered regions, and nurse skill levels to test how SBAM works under different conditions that mimic real life in the following sections.

### 4.4.2 Clustered Service Area

We assume that patient requests arrive equally likely from a region in all simulation experiments that we carried out so far. However, the number of patient request arriving from one region can be higher than from another region. Maybe some regions are slightly or not populated. To be able to test our algorithm under those conditions, we cluster patient requests in three rectangular subregions with given coordinates $X \in [0, 60]$ and $Y \in [0, 60]$, as shown in Figure 4.5. Two different cluster types are proposed. Patient requests arrive only from these subregions equally likely in Cluster 1. In Cluster 2, 70% of patient requests arise from those subregions

Table 4.15: Total daily visits, travel times per visit, acceptance rates, and the percentage changes between daily visits and travel times of SBAM and DHM for 12 nurses according to day set 2

| Interarrival time | Method | Total daily visit | % | Travel time per visit | % | Acceptance rates (%) |
|---|---|---|---|---|---|---|
| 255 | SBAM | 41.64 | 0 | **12.27** | -50.2 | 100 |
| | DHM | 41.33 | | 24.64 | | 100 |
| 150 | SBAM | **67.82** | 3.8 | **13.87** | -41.9 | 97 |
| | DHM | 65.35 | | 23.90 | | 94 |
| 100 | SBAM | **78.61** | 10.3 | **13.85** | -41.7 | 77 |
| | DHM | 71.25 | | 23.77 | | 71 |

Table 4.16: Total daily visits, travel times per visit, acceptance rates, and the percentage changes between daily visits and travel times of SBAM and DHM for 24 nurses according to day set 2

| Interarrival times | Method | Total daily visit | % | Travel time per visit | % | Acceptance rates (%) |
|---|---|---|---|---|---|---|
| 100 | SBAM | 105.08 | 0 | **8.58** | -60.3 | 100 |
| | DHM | 105.66 | | 21.63 | | 100 |
| 75 | SBAM | **139.49** | 3.1 | **9.53** | -55.1 | 99 |
| | DHM | 135.27 | | 21.24 | | 97 |

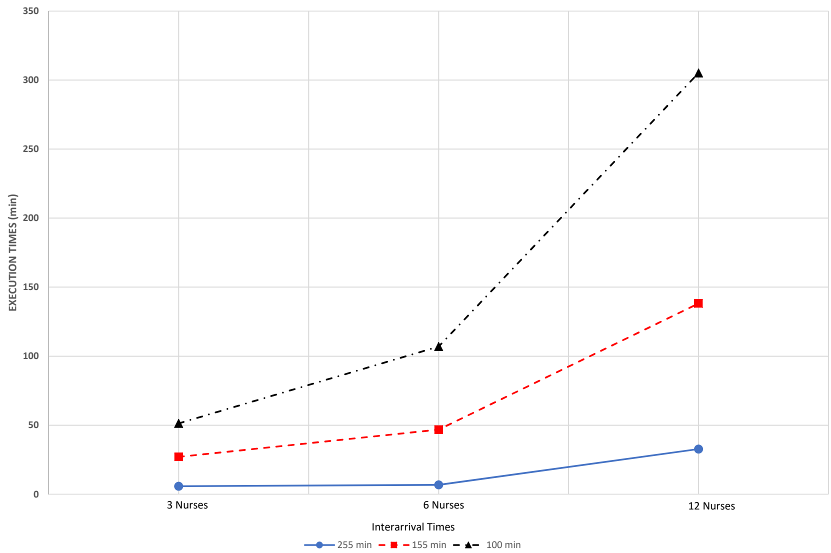



Figure 4.4: Execution times (minute) of 30 trials for different number of nurses and interarrival times

while 30% arrive from the remaining area of the whole service region. We use three interarrival times (340,255,150) and three nurses who are located at (10,10), (30,30), and (40,50) in a square region.

Table 4.17 reports the result of this evaluation for the case of 3 nurses and day set 1. When the demand intensity is low (interarrival time is 340 minutes) for each regional demand case, there are no meaningful differences between average daily visits of the two methods since acceptance rates for both approaches are 100%. In other words, nurses have sufficient capacity to accept all patients. However, when we look at travel time per visit, SBAM reduces it by 24%, 51%, and 34%. When the demand intensity is higher (interarrival times 255 or 150 minutes), SBAM accommodates more total daily visits than DHM. Particularly if patient demand arrives only from subregions (Cluster 1), increases in total daily visits are higher. SBAM improves total daily visits by 11% in the high demand scenario (150 minutes interarrival time). It is important to emphasize that acceptance rate is around 85% in that case. When demand increases, the gap between average daily visits of the two methods increases as well. All differences between results except those for average

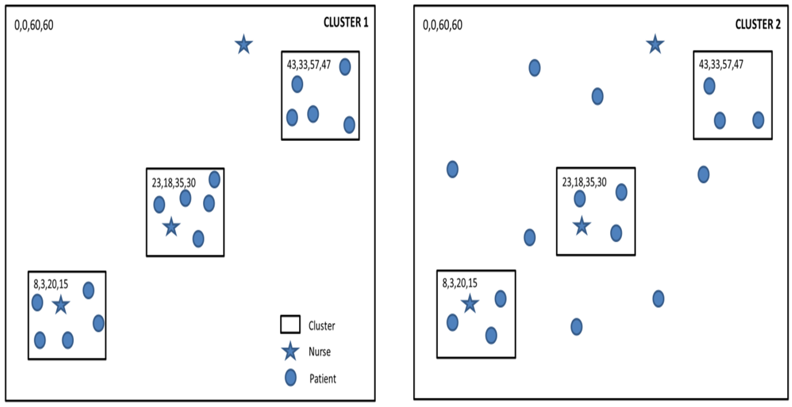



Figure 4.5: Spatial distribution of clusters and nurses

daily visits for 340 minutes are statistically significant. Similarly SBAM provides equal daily visit distribution to nurses while the differences between daily visits of nurses sometimes exceed 2 visits per day in DHM.

Table 4.18 reports the result of this evaluation for the case of 3 nurses and day set 2. Because we can assign patients to only some specific day combinations, average daily visits decreases compared to day set 1. However, we can observe the same overall picture as before. In low demand scenarios, SBAM reduces travel time per visit significantly for uniform and clustered demand scenarios even though total daily visits of both methods are similar. Under high demand (150 minutes), SBAM improves average daily visits by 11%, 19%, and 16% for each regional demand scenario respectively. Again, all differences between results except those for average daily visits for 340 minutes are statistically significant.

Another important issue for HHC providers is to ensure fair workloads for their caregivers. It is not acceptable for a member of staff to have to substantially work more than others over a prolonged period of time without any compensation. In the literature, some studies are devoted to balance workloads of workers [Hertz and Lahrichi, 2009]. Although our objective is to maximise total daily visits, we should examine workloads of nurses explicitly. We find a visit range for each trial, which is

Table 4.17: Total daily visits, travel times per visit, and the percentage changes between daily visits and travel times of SBAM and DHM for three nurses according to day set 1

| Demand | Interarrival times | Method | Range | Total daily visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|---|
| UNIFORM | 340 | SBAM | 1.06 ±0.15 | 15.69 | 0.0 | **20.14** | -24.0 |
| | | DHM | 1.03 ±0.14 | 15.88 | | 26.52 | |
| | 255 | SBAM | 0.36 ±0.08 | **20.57** | 2.9 | **19.48** | -24.5 |
| | | DHM | 0.38 ±0.06 | 19.98 | | 25.81 | |
| | 150 | SBAM | 0.19 ±0.03 | **25.95** | 9.7 | **17.83** | -29.7 |
| | | DHM | 0.17 ±0.04 | 23.66 | | 25.36 | |
| CLUSTER1 | 340 | SBAM | 0.74 ±0.17 | 15.70 | 0.0 | **8.95** | -51.3 |
| | | DHM | 2.47 ±0.22 | 15.83 | | 18.38 | |
| | 255 | SBAM | 0.58 ±0.13 | **21.22** | 2.3 | **8.38** | -50.5 |
| | | DHM | 1.34 ±0.13 | 20.74 | | 16.93 | |
| | 150 | SBAM | 0.26 ±0.05 | **29.55** | 9.9 | **8.56** | -46.0 |
| | | DHM | 0.40 ±0.07 | 26.88 | | 15.86 | |
| CLUSTER2 | 340 | SBAM | 0.86 ±0.15 | 15.73 | 0.0 | **14.90** | -33.9 |
| | | DHM | 1.60 ±0.15 | 15.83 | | 22.54 | |
| | 255 | SBAM | 0.50 ±0.08 | **20.91** | 2.6 | **14.48** | -32.8 |
| | | DHM | 1.26 ±0.10 | 20.38 | | 21.54 | |
| | 150 | SBAM | 0.30 ±0.06 | **27.67** | 9.5 | **12.97** | -36.1 |
| | | DHM | 0.38 ±0.07 | 25.27 | | 20.30 | |

Table 4.18: Total daily visits, travel times per visit, and the percentage changes between daily visits and travel times of SBAM and DHM for three nurses according to day set 2

| Demand | Interarrival | Method | Range | Total Daily Visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|---|
| UNIFORM | 340 | SBAM | 0.63 ±0.11 | 14.39 | 0.0 | **23.83** | -18.6 |
| | | DHM | 1.83 ±0.08 | 14.36 | | 29.29 | |
| | 255 | SBAM | 0.59 ±0.10 | **17.36** | 6.1 | **22.80** | -22.3 |
| | | DHM | 2.03 ±0.07 | 16.36 | | 29.36 | |
| | 150 | SBAM | 0.26 ±0.06 | **22.22** | 10.8 | **20.58** | -28.9 |
| | | DHM | 1.41 ±0.12 | 20.06 | | 28.95 | |
| CLUSTER1 | 340 | SBAM | 0.50 ±0.10 | 15.26 | 0.0 | **10.73** | -40.2 |
| | | DHM | 1.58 ±0.21 | 15.26 | | 17.96 | |
| | 255 | SBAM | 0.54 ±0.10 | **19.19** | 7.6 | **10.98** | -42.2 |
| | | DHM | 1.89 ±0.14 | 17.84 | | 18.99 | |
| | 150 | SBAM | 0.42 ±0.07 | **25.82** | 18.6 | **9.65** | -48.9 |
| | | DHM | 1.71 ±0.09 | 21.77 | | 18.88 | |
| CLUSTER2 | 340 | SBAM | 0.40 ±0.08 | 14.75 | 0.0 | **17.21** | -27.8 |
| | | DHM | 1.67 ±0.14 | 14.85 | | 23.84 | |
| | 255 | SBAM | 0.56 ±0.09 | **18.11** | 6.5 | **16.67** | -31.1 |
| | | DHM | 2.02 ±0.09 | 17.00 | | 24.20 | |
| | 150 | SBAM | 0.33 ±0.05 | **24.11** | 15.7 | **14.20** | -40.6 |
| | | DHM | 1.64 ±0.10 | 20.84 | | 23.91 | |

the difference between maximum and minimum daily visits of nurses. "Range" in Table 4.17 and 4.18 show average ranges and confidence intervals with 95% confidence level. The only case when the difference between daily visits of nurses clearly exceed one visit is day set 1-low demand-uniform scenario. In other scenarios, particularly high demands, differences between daily visits of nurses are very close to zero. On the other hand, DHM causes unbalanced workloads in many scenarios. Differences between daily visits of nurses exceed one visit and sometimes two visits.

### 4.4.3 Same Depots

In the HHC problem, nurses usually start the daily routine from their home and return to their home after finishing all visits. In our case and simulation settings, we also accept this common situation. However, some HHC companies can request their workers to arrive at the work office of the company at the beginning and end of each day. For example, if nurses should pick up some medicines or necessary appliances before visits or the company requests daily discussions and reports about visits, this condition can be taken into consideration. In this case, we have to consider only one depot instead of multi depots where each represents the home of a nurse. We assume that there are three nurses that have to arrive and return to a care office located at the centre of the service area (30,30). Nurses should start daily visits from the office at 8.00 and return to the office at 16.30. We ignore travel times between the office and nurses' homes.

Table 4.17 and 4.18 represent that nurses start daily visits from their homes. If they are compared with Table 4.19 and 4.20, total daily visits are slightly different while travel times per visit increase somewhat even though the depot is located at the centre of the service area.

As in the case of nurses starting from their home, SBAM outperforms DHM in terms of total daily visits and travel times per visit under medium and high demands. SBAM provides up to 31% shorter travel times and 14% higher total daily visits compared to DHM under high demands. The differences under low demands

Table 4.19: Total daily visits and travel times per visit for SBAM and DHM when employing three nurses for day set 2

| Demand | Interarrival times | Method | Range | Total daily visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|---|
| UNIFORM | 340 | SBAM | 0.90 ±0.14 | 14.24 | 0 | **25.08** | -10.81 |
| | | DHM | 1.97 ±0.06 | 14.47 | | 28.12 | |
| | 255 | SBAM | 0.96 ±0.14 | **17.19** | 4.42 | **24.12** | -15.29 |
| | | DHM | 2.15 ±0.08 | 16.46 | | 28.47 | |
| | 150 | SBAM | 0.27 ±0.05 | **22.10** | 10.75 | **21.55** | -23.38 |
| | | DHM | 1.54 ±0.09 | 19.96 | | 28.12 | |
| CLUSTER1 | 340 | SBAM | 0.93 ±0.17 | 15.03 | 0 | **14.56** | -12.59 |
| | | DHM | 0.98 ±0.13 | 15.13 | | 16.65 | |
| | 255 | SBAM | 1.07 ±0.19 | **19.04** | 5 | **14.11** | -20.35 |
| | | DHM | 2.35 ±0.10 | 18.13 | | 17.72 | |
| | 150 | SBAM | 0.39 ±0.08 | **25.85** | 13.61 | **12.46** | -31.18 |
| | | DHM | 1.74 ±0.11 | 22.75 | | 18.10 | |
| CLUSTER2 | 340 | SBAM | 0.99 ±0.15 | 14.65 | 0 | **19.92** | -14.30 |
| | | DHM | 0.96 ±0.13 | 14.82 | | 23.25 | |
| | 255 | SBAM | 0.90 ±0.14 | **18.30** | 5.74 | **19.09** | -19.20 |
| | | DHM | 2.34 ±0.08 | 17.31 | | 23.62 | |
| | 150 | SBAM | 0.39 ±0.07 | **24.32** | 13.27 | **16.59** | -28.97 |
| | | DHM | 1.45 ±0.10 | 21.47 | | 23.35 | |

Table 4.20: Total daily visits and travel times per visit for SBAM and DHM when employing three nurses for day set 1

| Demand | Interarrival times | Method | Range | Total daily visits | % | Travel time per visit | % |
|---|---|---|---|---|---|---|---|
| UNIFORM | 340 | SBAM | 1.05 ±0.18 | 15.58 | 0 | **22.40** | -15 |
| | | DHM | 0.53 ±0.12 | 15.50 | | 26.34 | |
| | 255 | SBAM | 0.38 ±0.06 | **20.18** | 2.59 | **21.39** | -16.86 |
| | | DHM | 0.30 ±0.07 | 19.67 | | 25.73 | |
| | 150 | SBAM | 0.23 ±0.03 | **25.16** | 6.47 | **19.23** | -23.62 |
| | | DHM | 0.17 ±0.04 | 23.63 | | 25.18 | |
| CLUSTER1 | 340 | SBAM | 1.19 ±0.20 | 15.63 | 0 | **13.99** | -20.96 |
| | | DHM | 0.98 ±0.19 | 15.49 | | 17.70 | |
| | 255 | SBAM | 0.72 ±0.18 | 20.59 | 0 | **12.70** | -23 |
| | | DHM | 0.58 ±0.16 | 20.40 | | 16.43 | |
| | 150 | SBAM | 0.43 ±0.05 | **28.25** | 4.78 | **11.45** | -26.68 |
| | | DHM | 0.27 ±0.05 | 26.96 | | 15.61 | |
| CLUSTER2 | 340 | SBAM | 1.05 ±0.20 | 15.52 | 0 | **19.02** | -14 |
| | | DHM | 0.72 ±0.13 | 15.40 | | 22.16 | |
| | 255 | SBAM | 0.79 ±0.14 | **20.39** | 1.55 | **17.56** | -17.34 |
| | | DHM | 0.38 ±0.08 | 20.08 | | 21.24 | |
| | 150 | SBAM | 0.46 ±0.06 | **26.70** | 6.23 | **15.03** | -26.07 |
| | | DHM | 0.22 ±0.05 | 25.13 | | 20.33 | |

are statistically not significant. Another issue as we mentioned before about DHM is unbalanced workloads. In some cases, a nurse can work almost 50% more than another nurse. It is quite doubtful that workers and companies would tolerate unfair workloads like this in the long term.

In this section, we tested the performance of our algorithm when nurses have to arrive at a central office before starting and after ending their daily visits. According to results, SBAM shows that it is still robust in this case. Furthermore, results show that even if the depot is located at the centre of the service area, the fact that nurses begin daily services from their home ensures shorter travel times and higher daily visits even though daily travel times between homes of nurses and their workplaces are ignored.

## 4.5 Nurse Districting Problem

One way to adapt single nurse approaches to the case of multiple nurses is to split a service region into several districts. Districting problems, also called territory design, territory alignment, zone, or sector design, are concerned with defining areas in a geographical region in order to distribute scarce sources into those areas effectively [Kalcsics, 2015]. Effectiveness depends on some criteria such as balance, contiguity, and compactness. Balance can be described in terms of workloads of workers and the number of customers. Contiguity and compactness are related to the geographical shape and boundaries of territories, and have effect on travel times [Kalcsics, 2015]. The districting problem has a broad range of application areas such as political, school, waste collection, police patrolling area districtings. The districting problem for HHC is simply how the shape and size of the subregions we have to define in order to minimise travel times, balance workloads of nurses, and maximise acceptance of patients. It is not very straightforward since demand and population fluctuations in subregions cause workload inequities between nurses. In addition, if more qualified nurses are scarce resources, we have to need overlapping areas. Finally, the territories are fixed and can not quickly react to the dynamically changing requests.

Besides multiple nurses routing and scheduling problem, we want to show in this study how average daily visits and travel times per visit are affected if we use DH and SBA for more than one nurse servicing in several territories as considered in the nurse districting problem against SBAM for same nurses servicing across the whole service region without any territory restriction.

In this study, we use idealistic settings which include a square service area, equal size subregions, and equal expected demand for each subregion. In real life, districting problems are complicated and there are many studies related to determination of optimal area size for each subregion by considering different constraints [Kalcsics, 2015].

### 4.5.1 Simulation Settings

We tested two different nurse sets with 2 and 4 nurses. In the first scenario, nurses are located in the centre of their service regions. The whole service region is divided into two equal size subregions, $X_1 \in [0, 30]$ and $Y_1 \in [0, 60]$ and $X_2 \in [30, 60]$ and $Y_2 \in [0, 60]$, and nurses are located at (15,30) and (45,30). For four nurses, the whole service region is divided into four equal size subregions, $X_1 \in [0, 30]$ and $Y_1 \in [0, 30]$, $X_2 \in [0, 30]$ and $Y_2 \in [30, 60]$, $X_3 \in [30, 60]$ and $Y_3 \in [0, 30]$, and $X_4 \in [30, 60]$ and $Y_4 \in [30, 60]$. Nurses are located at (15,15), (15,45), (45,15), and (45,45). In the second scenario, nurses are not located in the centre of their service regions. For two nurses, they are located at (5,10) and (60,25), and four nurses, their locations are (5,10), (55,30), (0,50), and (60,25). Interarrival times between patient arrivals are 255 and 510 minutes for two nurse and 100 and 200 minutes for four nurse cases. The expected demand is divided equally among subregions. We compare this case with the case where nurses can service the whole service area without any restriction. Of course, the overall area and locations of nurses are identical. Moreover, we also test DH performance for the case that nurses only service in their subregions and the case that they can visit patient in the whole service area (DHM). To understand whether or not differences between average daily visits of both approaches are statistically significant, we use t-test and all results are statistically different.

### 4.5.2 Results

Tables 4.21 and 4.22 show average daily visits and travel times per visit for SBA, DH, DHM and SBAM. SBAM clearly increases average daily visits compared to SBA. Although differences are slightly higher for 2 nurses, SBAM allows to schedule at least one additional daily visit for four nurses case. As it is expected, districting service area reduces travel times per visit. For four nurses, the fact that nurses travel in the whole area increases travel times per visit by up to 23% while average daily visits rise by around 5%. It is considerable to sacrifice the small amount of visits in order to reduce long travel times. Note that nurses are located at the centre of their subregions and the demand in each subregion is equal in Scenario 1. Furthermore, DH increases average daily visits when nurses service in the whole area rather than only their own subregions as seen in Table 4.21. However, under high demand and 4 nurses, districting areas gives better results. The most important reason is that DHM fails to balance the workload of nurses. Therefore, once workload of a nurse is quickly filled, patients are assigned to nurses whose tours are not suitable. This causes higher travel times and fewer visits. Table 4.22 clearly shows quite long travel times when four nurses service in the whole region under high demand compared to their assignment to small regions.

Table 4.21: Average daily visits for SBA, DH, DHM and SBAM in Scenario 1

| | Interarrival times | DH | SBA | DHM | SBAM |
|---|---|---|---|---|---|
| 2 Nurses | **510** | 8.58 | 8.99 | 9.17 | **9.21** |
| | **255** | 12.06 | 12.93 | 12.25 | **13.49** |
| 4 Nurses | **200** | 21.26 | 22.02 | 22.14 | **23.08** |
| | **100** | 29.22 | 31.34 | 28.57 | **32.56** |

Although it is desirable that nurses are located close to the middle of their service

Table 4.22: Travel times per visit for SBA, DH, DHM and SBAM in Scenario 1

| | Interarrival times | DH | SBA | DHM | SBAM |
|---|---|---|---|---|---|
| 2 Nurses | **510** | 24.10 | **23.90** | 30.14 | 26.63 |
| | **255** | 22.39 | **19.58** | 29.81 | 24.20 |
| 4 Nurses | **200** | 15.40 | **15.37** | 27.91 | 18.97 |
| | **100** | 14.54 | **13.66** | 27.65 | 16.45 |

regions, it is not always possible. Scenario 2 represents a situation where nurses are not located close to the middle of their service areas. Table 4.23 and 4.24 represent average daily visits and travel times per visit for SBA and SBAM in Scenario 2. Although SBAM increases average daily visits in each case as in Scenario 1, increments are higher compared to results in Table 4.21. Moreover, changing location of nurses affects results in SBA more than SBAM. For example, average daily visits of nurses by SBA decrease around 10% while visits by SBAM decrease only 2%. On the other hand, DH shows the same pattern as in Scenario 2. When there are four nurses and the demand is high, assigning nurses to subregions increases average daily visits and lessens travel times per visit.

Table 4.23: Average daily visits for SBA, DH, DHM, and SBAM in Scenario 2

| | Interarrival times | DH | SBA | DHM | SBAM |
|---|---|---|---|---|---|
| 2 Nurses | **510** | 8.27 | 8.58 | 9.02 | **9.06** |
| | **255** | 11.76 | 12.38 | 12.04 | **13.46** |
| 4 Nurses | **200** | 20.57 | 21.80 | 21.90 | **22.83** |
| | **100** | 28.69 | 30.17 | 28.15 | **32.00** |

Overall, results show that considering the whole service area, demand over the whole service area and all nurses at the same time when routing and scheduling provides better results in terms of average daily visits in each scenario. Moreover, SBAM seems more robust against changes of nurses' locations and expected de-

Table 4.24: Travel times per visit for SBA, DH, DHM, and SBAM in Scenario 2

| | Interarrival times | DH | SBA | DHM | SBAM |
|---|---|---|---|---|---|
| 2 Nurses | **510** | 26.65 | **26.61** | 32.77 | 29.26 |
| | **255** | 24.41 | **21.65** | 32.02 | 25.71 |
| 4 Nurses | **200** | 17.95 | **16.01** | 29.64 | 21.00 |
| | **100** | 16.51 | **14.95** | 29.26 | 18.19 |

mands. It is notable that Scenario 1, with equal size territories, equal demand and central nurse locations, is an idealistic condition for a districting problem. Therefore, average daily visits and travel times per visit are near the best. When some conditions change as Scenario 2, results also deteriorate.

## 4.6 Qualification Levels

So far, we assumed that all nurses are homogeneous in terms of their skill level. The assumption may be justifiable in real life since a company can be specialized for only one type of nursing service. However, companies often provide a range of services to patients. If all workers in a company are just assigned to tasks that exactly match their skill level, we can consider this problem as homogeneous nurses that we have constructed before. All we need to do is to find demand for all different tasks, distinguish nurses in terms of their qualifications, and construct schedules and routes for them separately. However, caregivers qualified for a particular level can also perform tasks of lower qualification levels. For example, if a company employs two different nurses, senior and junior, senior nurses can be asked to perform some tasks that junior nurses can perform due to lack of junior nurses at that time or lack of demand for tasks that only senior nurses are qualified for. On the other hand, junior nurses are not allowed to perform some tasks that require higher qualifications. Of course, hourly cost of more skilled workers' qualifications is higher than that of less skilled workers. Therefore, decision makers have to take this into account when

Table 4.25: Total daily visits, travel times per visit, and acceptance rates for 3 assignment strategies and DHM under 60/40% demand estimation for Type 1 and Type 2 patients

| Scenario | Total daily visit | Travel time per visit | TotalAcc | Type1Acc | Type2Acc |
|---|---|---|---|---|---|
| Mixed | 51.82 | 19.70 | 0.75 | 0.95 | 0.47 |
| Homo | 52.21 | 22.51 | 0.76 | 0.84 | 0.65 |
| Prior | 51.97 | 22.0 | 0.76 | 0.91 | 0.55 |
| DHM | 49.29 | 25.96 | 0.73 | 0.90 | 0.47 |

maximising the number of visits.

In this case, we examine different assignment strategies, "Mixed", "Homo", and "Prior". "Mixed"strategy represents mixed assignments where a patient can be assigned to a nurse if the nurse has sufficient skill level. "Homo"is a homogeneous assignment where patients can only be assigned to nurses who are exactly matched in terms of skill levels. "Prior"gives priority to patients who need high skill services when assigning them in the scenario generation phase and we can assign patients who need lower skill to higher skilled nurses if and only if there is no patient who needs higher skilled nurses in the scenario. Finally, we demonstrate results of DHM by considering only mixed assignments since SBAM has given better results compared to DHM under homogeneous assignments that we have tested in previous sections.

We test three assignment strategies for two different types of patients (Type 1 and Type 2) according to their need of nurses, junior and senior. Moreover, we have two different demand estimations for patients. First, we assume that 60% of patients need nurses qualified at least junior level and others need senior nurses. Second, 80% of patients need nurses qualified at least junior level. There are 9 nurses, 6 junior and 3 senior. Locations of nurses are generated randomly from the service area and remain the same for each trial.

In Tables 4.25 and 4.26, we show some results where "TotalAcc"refers to the proportion of patients accepted. "Type1Acc and Type2Acc"represent acceptance rates

Table 4.26: Total daily visits, travel times per visit, and acceptance rates for 3 assignment strategies and DHM under 80/20% demand estimation for Type 1 and Type 2 patients

| Scenario | Total daily visit | Travel time per visit | TotalAcc | Type1Acc | Type2Acc |
|---|---|---|---|---|---|
| Mixed | 54.28 | 20.02 | 0.79 | 0.85 | 0.54 |
| Homo | 50.41 | 21.83 | 0.74 | 0.69 | 0.95 |
| Prior | 54.09 | 20.73 | 0.79 | 0.84 | 0.61 |
| DHM | 51.80 | 24.90 | 0.76 | 0.82 | 0.54 |

of Type 1 and Type 2 patients, respectively. According to Table 4.25, “Homo”provides the highest acceptance rate for Type 2 patients while keeping total daily visits and total acceptance rate at the same level (even slightly better but not statistically significant) with other strategies. This is expected since we know that although 40% of patients need senior nurses, only 33% of nurses have this skill level. Thus, dedicating senior nurses to Type 2 patients causes the highest acceptance rate for those patients. The second best, “Prior”, shows that giving absolute priority Type 2 patients and assign Type 1 patients to senior nurses only after there is no Type 2 patient in the scenario seems a good strategy since we do not know exact demands, but we can guess more or less which type of service has a higher demand. Therefore, “Prior”looks applicable under different demands and when acceptance of some kind of patients is more valuable.

Table 4.26 shows total daily visits, travel times per visit, and acceptance rates for 3 assignment strategies and DHM when 80% of patients need junior nurses and 20% need senior nurses. In this setting, total daily visits in “Homo”is lower than other strategies since dedicating all senior nurses to only Type 2 patients cause ineffective utilization. Although the difference between total daily visits of “Prior”and “Mixed”is not statistically significant, the acceptance rate of “Prior”for Type 2 patients is higher than “Mixed”. However, when comparing the acceptance rates of Type 2 patients in “Prior”with “Homo”, the gap is massive since the possibility of accepting Type 1 patients during scenario generation phase is high even if the algorithm gives priority to Type 2 patients. Travel times per visit in "Prior"are slightly higher than "Mixed", but lower than "Homo"and DHM. This shows that we have to accept longer travel times to accept less suitable requests for tours.

Although "Prior"where the algorithm gives priority to patients who demand higher skilled nurses in the scenario generation phase provides robust results depending on two different demand structures, mixed and homogeneous assignments can be reasonable according to targets of companies. For example, according to acceptance rates in Table 4.26, if the company charges Type 1 patient service hour 100$, the most profitable strategy is mixed assignment if hourly price for Type 2 patient is up to 108$. If hourly price for Type 2 patient is considered between 108$ and 176$, "Prior"gives the highest profit. After 176$, the best strategy is to use homogeneous assignment. This is valid if the company only consider profit maximisation. If the aim is to visit as many patient as possible, the mixed assignment can be chosen.

## 4.7 Patient Dependent Service Times

As we explained in Section 3.5, some patients' tasks can take longer than others. Therefore, employing a variety of service times can be more reasonable if one considers real cases. We test two different service time distributions and cost factors similar to those as in Section 3.5 for three nurses. Visit times can be 15, 30, and 45 minutes with probability 0.30, 0.35, and 0.35 in the first scenario and 0.10, 0.30, and 0.60 respectively in the second scenario. Three nurses who are located at (10,10), (30,30), and (40,50) in a square region $X \in [0, 60]$ and $Y \in [0, 60]$ are employed in each scenario and we test the interarrival time with 150 minutes instead of 340 minutes to be able to observe more hectic schedules. Patient requests are uniformly distributed across the service area. We use exactly the same cost factors that are explained in Section 3.5.

Tables 4.27 and 4.28 show test results in terms of total daily service durations,

Table 4.27: Comparison of SBAM with two different strategies, and DHM under patient dependent service times for day set 1

| | Interarrival time | Strategy | Duration | Daily visits | Travel time per Visit | Acceptance rate(%) |
|---|---|---|---|---|---|---|
| Scenario 1 | 255 | Travel/Service | **626.85** | **20.43** | **20.23** | **97.38** |
| | | Travel | **625.24** | **20.39** | **20.10** | **97.21** |
| | | DHM | 597.14 | 19.76 | 25.12 | 94.92 |
| | 150 | Travel/Service | **774.04** | **24.82** | **19.62** | **73.77** |
| | | Travel | **768.85** | **25.02** | **19.37** | **74.31** |
| | | DHM | 684.44 | 23.17 | 25.45 | 72.22 |
| Scenario 2 | 255 | Travel/Service | **731.41** | **19.50** | **21.72** | **94.96** |
| | | Travel | **730.50** | **19.51** | **21.71** | **95.05** |
| | | DHM | 676.69 | 18.33 | 27.78 | 90.87 |
| | 150 | Travel/Service | **872.73** | **23.07** | **20.30** | **69.17** |
| | | Travel | **869.84** | **23.21** | **19.85** | **69.48** |
| | | DHM | 750.01 | 20.14 | 26.87 | 64.10 |

Table 4.28: Comparison of SBAM with two different cost strategies, and DHM under patient dependent service times for day set 2

| | Interarrival time | Strategy | Duration | Daily visit | Travel time per visit | Acceptance rate(%) |
|---|---|---|---|---|---|---|
| Scenario 1 | 255 | **Travel/Service** | **522.94** | **16.98** | **22.27** | **84.19** |
| | | **Travel** | **523.08** | **17.00** | **22.09** | **84.27** |
| | | **DH** | 485.75 | 16.01 | 28.65 | 81.65 |
| | 150 | **Travel/Service** | **662.91** | **21.25** | **21.19** | **64.61** |
| | | **Travel** | **658.47** | **21.40** | **20.59** | **65.08** |
| | | **DH** | 580.11 | 19.30 | 26.29 | 64.98 |
| Scenario 2 | 255 | **Travel/Service** | **616.79** | **16.45** | **23.01** | **81.92** |
| | | **Travel** | **616.31** | **16.47** | **22.97** | **81.96** |
| | | **DH** | 553.97 | 15.01 | 29.70 | 77.20 |
| | 150 | **Travel/Service** | **747.43** | **19.76** | **21.54** | **60.57** |
| | | **Travel** | **744.96** | **19.88** | **21.09** | **60.85** |
| | | **DH** | 635.68 | 17.07 | 27.75 | 57.68 |

daily visits, travel times per visit, acceptance rates for two day sets. First, no matter what cost factor is used, SBAM always provides longer daily service durations, higher daily visits, and shorter travel times per visit than DHM. If the demand is high, using "*Travel/Service*" cost factor when constructing tours in scenario phase ensures slightly better results compared to using "*Travel*".

## 4.8 Relaxation of Visit Time and Nurse Continuity

As we mentioned in Section 3.6, service continuity, which is that same nurse visits a patient at same days/times during his/her service horizon is one of constraints in this study. However, companies maybe interested in how this restriction affects costs in terms of travel times per visit and total daily visits. In Section 3.6, we tested how average daily visits and travel times per visit are changed if weekly visits are made at different times each week during patients' service horizons. In this section, we consider not only flexible visit times but also the flexible nurse, different nurses can service a patient during his/her service horizon which represented in "*Flex Time/Nurse*" column. Whenever a patient arrives, he or she is only informed whether or not his or her request is accepted and visit days and times for only next week. For the remaining episode of care, weekly schedules are made at the beginning of the previous week. All accepted patients' visits are optimally scheduled by the cheapest insertion heuristic at the beginning of the week. SBAM is only used for scheduling new patients' next week visits. We have three nurses who are located at (10,10), (30,30), and (40,50) in a square region $X \in [0, 60]$ and $Y \in [0, 60]$.

Tables 4.29 and 4.30 show total daily visits and travel times per visit for three nurses and two day sets. Under the low demand, total daily visits are slightly better if we violate service continuity while travel times per visit decrease between 5% and 10% for day set 1. However, total daily visits increase around 14% while travel times per visit falls by roughly 9% for day set 2. The flexibility of both visit times and nurses significantly decreases travel times per visit compared to the flexibility of only visit times.

Table 4.29: Total daily visits under strict and flexible assignments for three nurses and two day sets

| Region | Times | Strict | Flex Time | % | Strict | Flex Time/Nurse | % |
|---|---|---|---|---|---|---|---|
| Day Set 1 | 255 | 20.57 | 20.91 | 0 | 20.57 | **21.01** | 2.15 |
| | 150 | 25.95 | **27.55** | 6.18 | 25.95 | **28.43** | 9.58 |
| Day Set 2 | 255 | 17.36 | **18.82** | 8.45 | 17.36 | **19.79** | 14.03 |
| | 150 | 22.22 | **24.87** | 11.91 | 22.22 | **24.96** | 12.34 |

Table 4.30: Travel time per visit under strict and flexible assignments for three nurses and two day sets

| Region | Times | Strict | Flex Time | % | Strict | Flex Time/Nurse | % |
|---|---|---|---|---|---|---|---|
| Day Set 1 | 255 | 19.48 | **18.36** | -5.74 | 19.48 | **17.06** | -12.42 |
| | 150 | 18.17 | **16.18** | -10.93 | 18.17 | **14.73** | -18.89 |
| Day Set 2 | 255 | 22.80 | **20.21** | -9.78 | 22.80 | **18.23** | -20.04 |
| | 150 | 20.58 | **18.04** | -12.32 | 20.58 | **17.22** | -16.33 |

For both a singe and multiple nurse cases, the flexibility of visit times and nurses provide more patient visits and shorter travel times under different scenarios. Therefore, companies and decision makers can offer patients cheaper services without the service continuity or expensive services with the service continuity.

## 4.9 Patient Preference and Pricing Policy

As we mentioned in Section 3.7, patients want to choose visit days and times during their service horizon. If we have more than one nurse, patients also want to select a nurse due to some reasons such as gender and language preferences. In this section, we examine how a nurse and visit days/times preferences of a patient affects the total visits under different demands and service horizons. The logic is similar to Section 3.7. First, we assign the patient to a nurse that he or she preferred and run the simulation during his or her service horizon. After that, let the algorithm assign the patient to the nurse. Finally, we count the number of visits for both cases and compare results. Note that randomly generated requests, other patients arrival times, locations, weekly visits frequencies, and etc. are same for both cases.

We have three nurses located at (0,0), (30,30), and (60,60) in a service region $X \in [0, 60]$ and $Y \in [0, 60]$. The patient located at (59,55) needs 3 weekly visits during his or her service horizon. We test three different situations. First, the patient chooses both a nurse and service days and times. It is represented as "Nurse/Time" in the tables. Next, the patient only selects a nurse and service days/times are assigned by the algorithm. This is represented as "Nurse" in the tables. Finally, the algorithm is allowed to choose the nurse, visit days, and times.

Table 4.31: Total number of visits for three cases under 4 week service horizon

| Times | Nurse/Time | SBAM | Difference | Nurse | SBAM | Difference |
|---|---|---|---|---|---|---|
| **255** | 290.47 | 292.87 | **2.40** | 291.47 | 292.87 | **1.40** |
| **150** | 395.93 | 408.50 | **12.57** | 400.3 | 408.50 | **8.20** |

Table 4.32: Total number of visits for three cases under 8 week service horizon

| Times | Nurse/Time | SBAM | Difference | Nurse | SBAM | Difference |
|---|---|---|---|---|---|---|
| **340** | 675.30 | 693.50 | **18.20** | 680.30 | 693.50 | **13.20** |
| **255** | 785.63 | 808.87 | **23.23** | 789.67 | 808.87 | **19.20** |
| **150** | 935.60 | 964.13 | **28.53** | 941.03 | 964.13 | **23.10** |

Table 4.31 and 4.32 show the total number of visits for three cases under 4 and 8 week service horizons. Because the acceptance rate is around 100% when the interarrival time is 340 minutes and the service horizon is 4 weeks, we do not test 340 minute interarrival time for 4 week service horizon. In our example, let us assume that the patient selects the nurse located at (0,0). The algorithm assigns the patient to nurse located at (60,60) as expected since location of the patient is quite close and suitable for that nurse when we start with an empty schedule. Results clearly indicate that selection of an inappropriate nurse in terms of the location causes too many visits in both short and long service horizons. Although the algorithm decreases the lost around 30% by finding more suitable times, the gap between the results of SBAM and results of the patient' preferences is still high compared to the case of algorithm selected nurse.

Table 4.33: Total number of visits under only weekly visit days and times selection

| | **4 weeks** | | | **8 weeks** | | |
|---|---|---|---|---|---|---|
| **Times** | **Preference** | **SBAM** | **Difference** | **Preference** | **SBAM** | **Difference** |
| **340** | ** | ** | ****** | 691.70 | 693.50 | **1.80** |
| **255** | 291.90 | 292.87 | **0.97** | 802.93 | 808.87 | **5.93** |
| **150** | 405.23 | 408.50 | **3.27** | 948.87 | 964.13 | **15.27** |

Table 4.33 shows results when the patient only selects service day and times while the nurse is assigned by the algorithm. It is clear that the gap between results significantly decreases if the patient is assigned to the nurse whose tour is suitable to

the location of patient. Although patients choose their own visit times, differences between the total visits are relatively low for the 4 week service horizon and low demand compared to the difference when the patient selects a nurse whose tour is not suitable for the location of the patient.

In this setting, we assign visit days and times of a patient into the schedule of a nurse whose tour is not suitable for the location of the patient by assuming preference of the patient. It turns out that this causes significant visit loss even though the algorithm finds the most suitable days and times. Note that results are ensured by specific conditions of nurses and patient locations, current schedules of nurses, and etc. Therefore, differences between total visits of preferences and the algorithm can be highly volatile. As we mentioned in Section 3.7, pricing policy should reimburse a company for its losses in terms of visits. Thus, if a patient wants to select a special nurse and/or days and times, he or she should be charged as much as the deviation between results of his or her preference and assignments of the algorithm.

As we discussed in Section 3.7, although it is clear that the preferences of patients can change the total daily visits and travel times per visit dramatically, some other factors such as workloads of nurses and weekly visit needs of patients also affect the total number of visits. To evaluate the algorithm under different scenarios, it is wise to run a long simulation where we apply above procedure for each patient arriving during the simulation horizon. We find difference between the total number of visits based on the preference of patient and assignments of SBAM for each patient. Summing up those differences, called "Extra visits" in Table 4.34, and the total number of visits in schedules of nurses should be more or less equal the total number of visits according to assignments of SBAM. Similar to the procedure in Section 3.7, we assign each patient to days, times, and nurses by randomly picking up from feasible days and times of a nurse. Only difference is that the nurse is also randomly selected among all available nurses. If there is no space for the patient, he or she is rejected. We assume that each patient is willing to choose one of feasible day and time combinations from the schedule of a nurse. In other words, patients cannot select previously scheduled days and times.

Table 4.34: Comparison of the preference based assignment with the assignments of SBAM in terms of total number of visits for a year simulation horizon

| Interarrival times | Preference | Extra visits | Total visits | SBAM | p-value |
|---|---|---|---|---|---|
| **340** | 4739 | 276 | 5014 | 4969 | 0.26 |
| **255** | 5690 | 957 | 6647 | 6567 | 0.15 |
| **150** | 6979 | 1377 | 8356 | 8305 | ** |

According to Table 4.34, it turns out that summing up the total number of visits based on preferences and the total differences, "Extra visits", approximately equal to the total number of visits based on the assignments of SBAM. According to p values, we can say that there are no statistical differences between results when interarrival times are 340 and 255 minutes. We can run only one simulation for the highest demand case while we run 20 replications for the other two since it is computationally very demanding (one simulation with a year time horizon takes approximately 60 hours). This is why we cannot conduct t-test and provide p value. In even one trial, total number of visits of both approaches are close.

Table 4.35: Average daily visits, travel times per visit, and acceptance rates according to preference based and assignments of SBAM

| | **Average daily visits** | | **Travel times per visit** | | **Acceptance rates (%)** | |
|---|---|---|---|---|---|---|
| **Times** | **Preference** | **SBAM** | **Preference** | **SBAM** | **Preference** | **SBAM** |
| **340** | 14.81 | 15.69 | 37.67 | 20.06 | 98 | 100 |
| **255** | 17.78 | 20.57 | 35.92 | 19.51 | 91 | 98 |
| **150** | 21.81 | 25.95 | 29.45 | 17.88 | 66 | 76 |

Table 4.35 shows average daily visits, travel times per visit, and acceptance rates according to preference based and assignments of SBAM. Although the acceptance rate is 100% when interarrival time is 340 minutes, SBAM provides 276 more visits

with around 60% less travel time per visit. When the interarrival time is 150 minutes, the preference based assignment decreases the acceptance rate by 10%.

Overall, results of one-year simulation confirm that our pricing policy based on difference between patients' preferences and the algorithm based assignments for each patient provides more or less same results with assignments of SBAM. Therefore, we can say that if a company charges a patient according to lost visits due to days, times, and the nurse preferences of the patient, it will not make a loss in the long term. Note that we ignore travel time cost and rejection possibility of a patient to feasible times and days we provide. Furthermore, as discussed in Section 3.7, decision makers can derive an average cost per patient preference based visit from the proportion of lost visits to the total visits in a year and charge each visit of a patient with a standard fee.

# Chapter 5

# Conclusion and Future Work

Because of increasing average life expectancy, chronic diseases, and insufficiency of healthcare facilities, home care is getting more and more crucial everyday. However, many people who need care cannot access home care services due to lack of care workers. Therefore, companies have to use their workers' time efficiently in the scheduling and routing process.

In this study, acceptance and assignment time decisions have to be made as soon as patients arrive, where dynamic perspective is taken into consideration. Although there are some studies providing solutions to this problem by using greedy algorithms in the literature, these algorithms do not consider future demand. We propose a Scenario Based Approach (SBA) which is based on generating several scenarios of future demand to see whether or not we can assign visits of the patient who is currently under consideration. A scenario includes number of randomly generated requests in terms of weekly demand and expected number of visits. The basic idea behind the algorithm is to run a number of simulations (scenarios) to see how many times the patient is assigned among all requests and which time slot the patient is scheduled frequently. Based on this information, we decide to accept or reject the patient and the time slots he or she is scheduled.

First, we develop and analyse two different approaches, Daily SBA (DSBA) and Weekly SBA (WSBA). The former is constructing tours based on daily demand and independently for each day. The latter is based on generating visits based on weekly demand and visit frequency of patients, and construct a week tour. The results are close to each other while the computational time for WSBA is significantly higher than DSBA's. Therefore, we test and compare DSBA to the distance and capacity heuristics. We construct a simulation model where patients' requests arrive exponentially. We make 6 trials based on two different size regions and 3 different interarrival times where each trial includes 30 replications. DSBA is clearly superior to the distance and capacity heuristics in each scenario based on average daily visits. However, travel times of our method are slightly higher under low-demand scenarios while DSBA provides significantly shorter travels at medium and high demands and larger areas. Particularly, we have significant improvements compared to other two methods under 1.5 and 2 requests per day for most of cases. Additionally, we also test our algorithm for special day combination a patient's visits can be assigned. Results show that DSBA provides better performance under all scenarios. We also test the condition that continuity of care constraint is violated. We reschedule weekly visits at the beginning of each week and show how this affects the average daily visits and travel times. Finally, we develop a new method to determine cost of a patient preference based assignment and how to charge by comparing it with the assignment of SBAM.

Next, we propose an improved algorithm for multi-nurse case since the previous SBA does not work properly if each nurse and the demand are considered independently during the scenario generation phase. Therefore, we modify SBA to be able to consider all nurses in the setting and expected demand at the same time when assigning requests into tours. This modification, which is called SBA for multiple nurses (SBAM), gives better results compared to results of previous SBA and the distance heuristic. On the other hand, we test how SBAM works based on all nurses service in the whole area versus SBA by assigning each nurse to a subregion. SBAM provides significantly higher average daily visits and acceptance rates with longer travel times. After that, we test SBAM against the distance heuristic for multiple nurses (DHM) for three, twelve, and twenty-four nurses under different demand structures, interarrival times and day sets. Results show that SBAM significantly increases the total daily visits and decreases travel times per visit compared to DHM. In high demands, average daily visits increase around 20% and travel times per visit are reduced by up to 50%. We test our algorithm if nurses are not homogeneous in terms of their skill levels. Three different strategies in terms of assignment structure are reviewed. The purpose is to propose different strategies to decision makers according to their needs or targets. We show how to choose a strategy in terms of hourly service prices of different patients. We also test how the violation of continuity of care and the patient preference based assignment affect daily visits, travel times, and acceptance rates. Results should support companies for their pricing policies depending on preferences of patients.

Overall, performance of our algorithm increases for higher demands and clustered areas compared to the greedy algorithms. Under considering a variety of scenarios such as different service times, service horizon, and violation of service continuity, our algorithm is superior to greedy algorithms. Although computational times for high number of nurses and demand significantly goes up for a year simulation horizon, they are still reasonable when considering assessment of each patient. Furthermore, if we consider nurse qualifications and a limited number of nurses whose tours are more suitable to location of a patient, computational times can be reduced notably.

We use data of [Bennett and Erera, 2011] derived from a HHC company in our experiments. Since many companies running all over the world under different restrictions and regulations, deriving data from these companies might not be so easy going and derived data can not be so suitable for our settings. For example, we assume that interarrival times of patients are exponentially distributed and next week demands are expected accordingly. However, interarrival times and demand can be remarkably vary over time. Furthermore, we ignore some common applications in HHC such as delay or cancel some visits, visit synchronisations, absence of nurses, etc. These factors are opportunities for future research.

In future research, I plan to extend this study in perspective of revenue management. In that case, patients are evaluated according to their profit margins that cover their visit durations, frequencies, service horizons, and type of their treatments. Furthermore, some patient visits need more than one nurse due to their complexities and optimising routing and scheduling of nurses under this constraint seems quite interesting as well as a challenge for future research. Lastly, I am interested in developing a software with a graphical user interface based on requirements of HHC companies. With this software, HHC companies easily schedules their nurses with minimum cost as well as considering special requirements and preferences of patients and caregivers.

# Bibliography

[AHRQ, 2007] AHRQ (2007). *2007 outcomes by patient and hospital characteristics for all discharges.* http://hcupnet.ahrq.gov/HCUPnet.jsp.

[Akjiratikarl et al., 2007] Akjiratikarl, C., Yenradee, P., and Drake, P. R. (2007). PSO-based algorithm for home care worker scheduling in the UK. *Computers & Industrial Engineering*, 53(4):559–583.

[Albareda-Sambola et al., 2014] Albareda-Sambola, M., Fernández, E., and Laporte, G. (2014). The dynamic multiperiod vehicle routing problem with probabilistic information. *Computers & Operations Research*, 48:31–39.

[Albers, 2003] Albers, S. (2003). Online algorithms: a survey. *Mathematical Programming*, 97(1-2):3–26.

[Albuquerque, 2010] Albuquerque, I. (2010). Why health care is going home.

[Angelelli et al., 2007] Angelelli, E., Grazia Speranza, M., and Savelsbergh, M. W. (2007). Competitive analysis for dynamic multiperiod uncapacitated routing problems. *Networks*, 49(4):308–317.

[Arabzadeh et al., 2016] Arabzadeh, E., Fatemi-Ghomi, S. M. T., and Karimi, B. (2016). A new mathematical model for multi period multi depot home health care routing scheduling problem. *Nursing Practice Today*, 3(4):140–151.

[Bard et al., 2014] Bard, J. F., Shao, Y., Qi, X., and Jarrah, A. I. (2014). The traveling therapist scheduling problem. *IIE Transactions*, 46(7):683–706.

[Begur et al., 1997] Begur, S. V., Miller, D. M., and Weaver, J. R. (1997). An integrated spatial dss for scheduling and routing home-health-care nurses. *Interfaces*, 27(4):35–48.

[Bennett and Erera, 2011] Bennett, A. R. and Erera, A. L. (2011). Dynamic periodic fixed appointment scheduling for home health. *IIE Transactions on Healthcare Systems Engineering*, 1(1):6–19.

[Bent and Van Hentenryck, 2004] Bent, R. W. and Van Hentenryck, P. (2004). Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6):977–987.

[Benzarti, 2012] Benzarti, E. (2012). *Home health care operations management: Applying the districting approach to home health care.* PhD thesis, Ecole Centrale Paris.

[Bertels and Fahle, 2006] Bertels, S. and Fahle, T. (2006). A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem. *Computers & Operations Research*, 33(10):2866–2890.

[Bertsimas et al., 2018] Bertsimas, D., Jaillet, P., and Martin, S. (2018). Online vehicle routing: The edge of optimization in large-scale applications.

[Braekers et al., 2016] Braekers, K., Hartl, R. F., Parragh, S. N., and Tricoire, F. (2016). A bi-objective home care scheduling problem: Analyzing the trade-off between costs and client inconvenience. *European Journal of Operational Research*, 248(2):428–443.

[Cappanera and Scutellà, 2014] Cappanera, P. and Scutellà, M. G. (2014). Joint assignment, scheduling, and routing models to home care optimization: A pattern-based approach. *Transportation Science*.

[Carello and Lanzarone, 2014] Carello, G. and Lanzarone, E. (2014). A cardinality-constrained robust model for the assignment problem in home care services. *European Journal of Operational Research*, 236(2):748–762.

[Census, 2011] Census (2011). *Projected population of the United States, by age and sex.* http://www.census.gov/ population/www/projections/usinterimproj.

[Chahed et al., 2009] Chahed, S., Marcon, E., Sahin, E., Feillet, D., and Dallery, Y. (2009). Exploring new operational research opportunities within the home care context: the chemotherapy at home. *Health Care Management Science*, 12(2):179–191.

[CMS, 2008] CMS (2008). *Home health quality initiatives. Technical report.* Centers for Medicare and Medicaid Services,7500 Security Boulevard, Baltimore MD, 21244.

[Decerle et al., 2016] Decerle, J., Grunder, O., El Hassani, A. H., and Barakat, O. (2016). A two-phases matheuristic for the home care routing and scheduling problem. *IFAC-PapersOnLine*, 49(12):1484–1489.

[Di Gaspero and Urli, 2014] Di Gaspero, L. and Urli, T. (2014). A CP/LNS approach for multi-day homecare scheduling problems. pages 1–15.

[Du et al., 2017] Du, G., Liang, X., and Sun, C. (2017). Scheduling optimization of home health care service considering patients priorities and time windows. *Sustainability*, 9(2):253.

[Duque et al., 2015] Duque, P. M., Castro, M., Sörensen, K., and Goos, P. (2015). Home care service planning. The case of Landelijke Thuiszorg. *European Journal of Operational Research*, 243(1):292–301.

[En-nahli et al., 2015] En-nahli, L., Allaoui, H., and Nouaouri, I. (2015). A multi-objective modelling to human resource assignment and routing problem for home health care services. *IFAC-PapersOnLine*, 48(3):698–703.

[Eveborn et al., 2006] Eveborn, P., Flisberg, P., and Rönnqvist, M. (2006). Laps care-an operational system for staff planning of home care. *European Journal of Operational Research*, 171(3):962–976.

[Fernandez et al., 1974] Fernandez, A., Gregory, G., Hindle, A., and Lee, A. (1974). A model for community nursing in a rural county. *Journal of the Operational Research Society*, 25(2):231–239.

[Fikar and Hirsch, 2017] Fikar, C. and Hirsch, P. (2017). Home health care routing and scheduling: A review. *Computers & Operations Research*, 77:86–95.

[Frifita et al., 2017] Frifita, S., Masmoudi, M., and Euchi, J. (2017). General variable neighbourhood search for home healthcare routing and scheduling problem with time windows and synchronized visits. *Electronic Notes in Discrete Mathematics*, 58:63–70.

[Guericke and Suhl, 2017] Guericke, D. and Suhl, L. (2017). The home health care problem with working regulations. *OR Spectrum*, pages 1–34.

[Hall, 2012] Hall, R. (2012). *Handbook of Healthcare System Scheduling*. Springer.

[Hertz and Lahrichi, 2009] Hertz, A. and Lahrichi, N. (2009). A patient assignment algorithm for home care services. *Journal of the Operational Research Society*, 60(4):481–495.

[Hewitt et al., 2016] Hewitt, M., Nowak, M., and Nataraj, N. (2016). Planning strategies for home health care delivery. *Asia-Pacific Journal of Operational Research*, 33(05):1650041.

[Hiermann et al., 2015] Hiermann, G., Prandtstetter, M., Rendl, A., Puchinger, J., and Raidl, G. R. (2015). Metaheuristics for solving a multimodal home-healthcare scheduling problem. *Central European Journal of Operations Research*, 23(1):89–113.

[Hvattum et al., 2006] Hvattum, L. M., Løkketangen, A., and Laporte, G. (2006). Solving a dynamic and stochastic vehicle routing problem with a sample scenario hedging heuristic. *Transportation Science*, 40(4):421–438.

[Ichoua et al., 2000] Ichoua, S., Gendreau, M., and Potvin, J.-Y. (2000). Diversion issues in real-time vehicle dispatching. *Transportation Science*, 34(4):426–438.

[Ichoua et al., 2006] Ichoua, S., Gendreau, M., and Potvin, J.-Y. (2006). Exploiting knowledge about future demands for real-time vehicle dispatching. *Transportation Science*, 40(2):211–225.

[International, 2015] International, A. D. (2015). Dementia statistics.

[Issabakhsh et al., 2018] Issabakhsh, M., Hosseini-Motlagh, S.-M., Pishvaee, M.-S., and Saghafi Nia, M. (2018). A vehicle routing problem for modeling home healthcare: a case study. *International Journal of Transportation Engineering*, 5(3):211–228.

[Issaoui et al., 2015] Issaoui, B., Zidi, I., Marcon, E., and Ghedira, K. (2015). New multi-objective approach for the home care service problem based on scheduling algorithms and variable neighbourhood descent. *Electronic Notes in Discrete Mathematics*, 47:181–188.

[Jaillet and Wagner, 2008] Jaillet, P. and Wagner, M. R. (2008). Generalized online routing: New competitive ratios, resource augmentation, and asymptotic analyses. *Operations research*, 56(3):745–757.

[Kalcsics, 2015] Kalcsics, J. (2015). Districting problems. In *Location science*, pages 595–622. Springer.

[Larsen et al., 2002] Larsen, A., Madsen, O., and Solomon, M. (2002). Partially dynamic vehicle routing models and algorithms. *Journal of the Operational Research Society*, 53(6):637–646.

[Larsen et al., 2008] Larsen, A., Madsen, O. B., and Solomon, M. M. (2008). Recent developments in dynamic vehicle routing systems. In *The vehicle routing problem: Latest advances and new challenges*, pages 199–218. Springer.

[Li et al., 2009] Li, J.-Q., Mirchandani, P. B., and Borenstein, D. (2009). A lagrangian heuristic for the real-time vehicle rescheduling problem. *Transportation Research Part E: Logistics and Transportation Review*, 45(3):419–433.

[Lin et al., 2017] Lin, C.-C., Hung, L.-P., Liu, W.-Y., and Tsai, M.-C. (2017). Jointly rostering, routing, and rerostering for home health care services: A harmony search approach with genetic, saturation, inheritance, and immigrant schemes. *Computers & Industrial Engineering.*

[Liu et al., 2013] Liu, R., Xie, X., Augusto, V., and Rodriguez, C. (2013). Heuristic algorithms for a vehicle routing problem with simultaneous delivery and pickup and time windows in home health care. *European Journal of Operational Research*, 230(3):475–486.

[Liu et al., 2017] Liu, R., Yuan, B., and Jiang, Z. (2017). Mathematical model and exact algorithm for the home care worker scheduling and routing problem with lunch break requirements. *International Journal of Production Research*, 55(2):558–575.

[López-Santana et al., 2016] López-Santana, E. R., Espejo-Díaz, J. A., and Méndez-Giraldo, G. A. (2016). Multi-agent approach for solving the dynamic home health care routing problem. In *Workshop on Engineering Applications*, pages 188–200. Springer.

[Lüers and Suhl, 2017] Lüers, D. and Suhl, L. (2017). Continuity between planning periods in the home health care problem. In *Operations Research Proceedings 2015*, pages 217–223. Springer.

[Manerba and Mansini, 2016] Manerba, D. and Mansini, R. (2016). The nurse routing problem with workload constraints and incompatible services. *IFAC-PapersOnLine*, 49(12):1192–1197.

[Mankowska et al., 2014] Mankowska, D. S., Meisel, F., and Bierwirth, C. (2014). The home health care routing and scheduling problem with interdependent services. *Health care management science*, 17(1):15–30.

[Marcon et al., 2017] Marcon, E., Chaabane, S., Sallez, Y., Bonte, T., and Trentesaux, D. (2017). A multi-agent system based on reactive decision rules for solving the caregiver routing problem in home health care. *Simulation Modelling Practice and Theory*, 74:134–151.

[Masmoudi and Mellouli, 2014] Masmoudi, M. and Mellouli, R. (2014). MILP for synchronized-mTSPTW: Application to home healthcare scheduling. In *Control, Decision and Information Technologies (CoDIT), 2014 International Conference on*, pages 297–302. IEEE.

[McKay et al., 1979] McKay, M. D., Beckman, R. J., and Conover, W. J. (1979). Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245.

[Milburn et al., 2012] Milburn, A. B., Mason, S. J., and Spicer, J. (2012). Characterizing the home health care supply chain. *Home Health Care Management & Practice*, page 1084822312439467.

[Mutingi and Mbohwa, 2013] Mutingi, M. and Mbohwa, C. (2013). Home healthcare staff scheduling: a taxonomic state-of-the-art review. In *2013 IEEE International Conference on Industrial Engineering and Engineering Management*, pages 1107–1111. IEEE.

[NAHC, 2016] NAHC (2016). *Stats and Facts in HHC.* https://www.nahc.org/advocacy-policy/research/home-care-hospice-facts-stats/.

[Nasir and Dang, 2017] Nasir, J. A. and Dang, C. (2017). An integrated model for home health care and patients cluster based care. In *Robotics and Automation Sciences (ICRAS), 2017 International Conference on*, pages 204–208. IEEE.

[Nguyen et al., 2015] Nguyen, T. V. L., Toklu, N. E., and Montemanni, R. (2015). Matheuristic optimization for robust home health care services. In *Proc. ICAOR*, page 2.

[NIH, 2015] NIH (2015). *World's older population grows dramatically.* https://www.nih.gov/news-events/news-releases/worlds-older-population-grows-dramatically.

[Pillac et al., 2013] Pillac, V., Gendreau, M., Guéret, C., and Medaglia, A. L. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1–11.

[Rasmussen et al., 2012] Rasmussen, M. S., Justesen, T., Dohn, A., and Larsen, J. (2012). The home care crew scheduling problem: Preference-based visit clustering and temporal dependencies. *European Journal of Operational Research*, 219(3):598–610.

[Redjem and Marcon, 2016] Redjem, R. and Marcon, E. (2016). Operations management in the home care services: A heuristic for the caregivers routing problem. *Flexible Services and Manufacturing Journal*, 28(1-2):280–303.

[Rest and Hirsch, 2016] Rest, K.-D. and Hirsch, P. (2016). Daily scheduling of home health care services using time-dependent public transport. *Flexible Services and Manufacturing Journal*, 28(3):495–525.

[Riazi et al., 2014] Riazi, S., Chehrazi, P., Wigström, O., Bengtsson, K., and Lennartson, B. (2014). A gossip algorithm for home healthcare scheduling and routing problems. *IFAC Proceedings Volumes*, 47(3):10754–10759.

[Ritzinger et al., 2016] Ritzinger, U., Puchinger, J., and Hartl, R. F. (2016). A survey on dynamic and stochastic vehicle routing problems. *International Journal of Production Research*, 54(1):215–231.

[Rodriguez et al., 2015] Rodriguez, C., Garaix, T., Xie, X., and Augusto, V. (2015). Staff dimensioning in homecare services with uncertain demands. *International Journal of Production Research*, 53(24):7396–7410.

[Secomandi, 2000] Secomandi, N. (2000). Comparing neuro-dynamic programming algorithms for the vehicle routing problem with stochastic demands. *Computers & Operations Research*, 27(11):1201–1225.

[Shao et al., 2012] Shao, Y., Bard, J. F., and Jarrah, A. I. (2012). The therapist routing and scheduling problem. *IIE Transactions*, 44(10):868–893.

[Shi et al., 2017] Shi, Y., Boudouh, T., and Grunder, O. (2017). A hybrid genetic algorithm for a home health care routing problem with time window and fuzzy demand. *Expert Systems with Applications*, 72:160–176.

[Steeg and Schroder, 2008] Steeg, J. and Schroder, M. (2008). A hybrid approach to solve the periodic home health care problem. In *Operations Research Proceedings 2007*, pages 297–302. Springer.

[Thomas, 2010] Thomas, B. W. (2010). Dynamic vehicle routing. *Wiley Encyclopedia of Operations Research and Management Science*.

[Tozlu et al., 2015] Tozlu, B., Daldal, R., Ünlüyurt, T., and Çatay, B. (2015). Crew constrained home care routing problem with time windows. In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 1751–1757. IEEE.

[Trautsamwieser and Hirsch, 2011] Trautsamwieser, A. and Hirsch, P. (2011). Optimization of daily scheduling for home health care services. *Journal of Applied Operational Research*, 3(3):124–136.

[Trautsamwieser and Hirsch, 2014] Trautsamwieser, A. and Hirsch, P. (2014). A branch-price-and-cut approach for solving the medium-term home health care planning problem. *Networks*, 64(3):143–159.

[Triki et al., 2014] Triki, N., Garaix, T., and Xie, X. (2014). A two-phase approach for periodic home health care planning. In *Automation Science and Engineering (CASE), 2014 IEEE International Conference on*, pages 518–523. IEEE.

[Wen et al., 2010] Wen, M., Cordeau, J.-F., Laporte, G., and Larsen, J. (2010). The dynamic multi-period vehicle routing problem. *Computers & Operations Research*, 37(9):1615–1623.

[Wilson and Colvin, 1977] Wilson, N. H. and Colvin, N. J. (1977). *Computer control of the Rochester dial-a-ride system*. Massachusetts Institute of Technology, Center for Transportation Studies.

[Wirnitzer et al., 2016] Wirnitzer, J., Heckmann, I., Meyer, A., and Nickel, S. (2016). Patient-based nurse rostering in home care. *Operations Research for Health Care*, 8:91–102.

[Yalçındağ et al., 2016a] Yalçındağ, S., Cappanera, P., Scutellà, M. G., Şahin, E., and Matta, A. (2016a). Pattern-based decompositions for human resource planning in home health care services. *Computers & Operations Research*, 73:12–26.

[Yalcindag et al., 2012] Yalcindag, S., Matta, A., and Sahin, E. (2012). Operator assignment and routing problems in home health care services. In *Automation Science and Engineering (CASE), 2012 IEEE International Conference on*, pages 329–334. IEEE.

[Yalçındağ et al., 2016b] Yalçındağ, S., Matta, A., Şahin, E., and Shanthikumar, J. G. (2016b). The patient assignment problem in home health care: using a data-driven method to estimate the travel times of care givers. *Flexible Services and Manufacturing Journal*, 28(1-2):304–335.

[Yang et al., 2000] Yang, W.-H., Mathur, K., and Ballou, R. H. (2000). Stochastic vehicle routing problem with restocking. *Transportation Science*, 34(1):99–112.

[Yuan et al., 2015] Yuan, B., Liu, R., and Jiang, Z. (2015). A branch-and-price algorithm for the home health care scheduling and routing problem with stochastic service times and skill requirements. *International Journal of Production Research*, 53(24):7450–7464.

[Zhan et al., 2015] Zhan, Y., Wang, Z., and Wan, G. (2015). Home care routing and appointment scheduling with stochastic service durations. *Available at SSRN*.

# Chapter 6

# Appendix

```
// DSBA
// A-year simulation horizon
public class Separate_Days {
static Random ran=new Random(1238); // Use the same seed for each
    experiment
static double totalTim=360; // Simulation horizon
static int mean=510; // Interarrival time
static int warmup=20; // Warm up period
static final int servicePe=4; // Episode of care
public static final double one=0.05; //Probability of arrival of a
    patient who needs a visit per week
public static final double two=0.35; //Probability of arrival of a
    patient who needs two visits per week
public static void main(String[] args) {
long startTime = System.currentTimeMillis();
ArrayList<String> data=new ArrayList<String>();
for (int q = 0; q < 30; q++) {  //The number of replications
double simTime=0;
double count=0;
double countReq=0;
double tour=0;
int weekCal=(int)(totalTim/5+servicePe);
serviceTot=0;
```

```
//Create weeks
Methodweek weeks[]=new Methodweek[weekCal];
for (int i = 0; i < weeks.length; i++) {
weeks[i]=new Methodweek(i);
}
while(simTime<totalTim*MethodDay.dayLength) {    //Termination
int weekNum=(int)(simTime/(MethodDay.dayLength*5));
String k="";
Request rex=new Request("X",ran.nextInt(Request.AreaX),ran.nextInt(
    Request.AreaY),0); // Generating a patient
countReq++;
int freq=frequency();  // Generating weekly visit frequency of the
    patient
k="Patient request arrives week "+(weekNum+1)+" "+timeCal(simTime)+"
    from "+rex.getX()+" "+rex.getY()+" with frequency "+freq;
// List the number of acceptances and time slots
ArrayList<entity> list=new ArrayList<entity>();
for (int i = 0; i < 5; i++) {
int z=0;
for (int j = 0; j < servicePe; j++) {
int t=decision(weeks[weekNum+j].getday(i).getOrder(),rex,simTime,(f))
    [0];
if (t<=0) {
z=Integer.MIN_VALUE;
break;
} else {
z+=t;
}}
int[] a=decision(weeks[weekNum].getday(i).getOrder(),rex,simTime,f);
list.add(new entity(weeks[weekNum].getday(i).getDaynumber(),z,a[1]));
}
//Find best visit days according to weekly visit frequency of the
    patient
ArrayList<entity> bests=new ArrayList<entity>();
ArrayList<Integer> maxfre=new ArrayList<Integer>();
for (int i = 0; i < freq; i++) {
```

```
int max=Integer.MIN_VALUE;
int ind=0;
for (int j = 0; j < list.size(); j++) {
if (list.get(j).getAcceptance()>=max) {
max=list.get(j).getAcceptance();
ind=j;
}}
bests.add(list.get(ind));
list.remove(ind);
}
//Check acceptance thresholds for each day are greater zero
int decision=0;
for (int i = 0; i < bests.size(); i++) {
if (bests.get(i).getAcceptance()<=0) {
decision=-1;
break;
}}
if (decision!=-1) {
k+=". It is accepted. Days and times: \n";
count++;
if (simTime>=warmup*MethodDay.dayLength ) {
serviceTot+=(freq*servicePe);
}
//Service weeks
for (int j = 0; j < servicePe; j++) {
int s=0;
int day=0;
// Visits in each week
for (int i = 0; i < freq; i++) {
s=bests.get(i).gettimeSlot();
day=bests.get(i).getDayNumber();
Request re=new Request(String.valueOf(count),rex.getX(),rex.getY(),1,s,
    s+(Request.PatService/Request.TimeSlot));
for (int t = 0; t < weeks[weekNum+j].getday(day).getOrder().size()-1; t
    ++) {
if (s>=weeks[weekNum+j].getday(day).getOrder().get(t).getEnd() && s<
```

```
weeks[weekNum+j].getday(day).getOrder().get(t+1).getStart()) {
weeks[weekNum+j].getday(day).getOrder().add(t+1,re);
if (j==0) {
k+=weeks[weekNum].getday(day).toString()+" "+re.startToString()+" "+re.
    endToString()+"\n";}
break;
}}}}}
else{k+=". It is rejected.";}
data.add(k);
simTime+=expDistr(mean);
}}
long endTime = System.currentTimeMillis();
System.out.println("That took " + (endTime - startTime) + "
    milliseconds");
}

// Decision block of DSBA
private static int[] decision(ArrayList<Request> or, Request re, double
     time, int day){
ArrayList<Integer> timeSlot=new ArrayList<Integer>();
int counter=0;
int week=(int)(time/2550.);
long scenarioSize=(int)(((2550)/(mean))*0.2*(1*Request.one+2*Request.
    two+3*(1-Request.one-Request.two)));
//Generating Scenarios
for (int p = 0; p < 75; p++) {
ArrayList<Request> ordersCopy=new ArrayList<Request>();
ArrayList<Request> scenario = new ArrayList<Request>();
//Getting previously assigned visits
for (int i = 0; i < or.size(); i++) {
ordersCopy.add(or.get(i));
}
scenario.add(re);
//Generating random requests
for (int j = 0; j < scenarioSize; j++) {
scenario.add(new Request("R",ran.nextInt(Request.AreaX),ran.nextInt(
```

```
    Request.AreaY),0));
}
ArrayList<Integer> forbInt=new ArrayList<Integer>();
//Constructing daily tour with the cheapest insertion heuristic
int a=0;
int loop=0;
while(a<15){
int indexofreq=0;
int indexofint=0;
double min=Integer.MAX_VALUE;
for (int j = 0; j < scenario.size(); j++) {
for (int j2 = 0; j2 < ordersCopy.size()-1; j2++) {
if (forbInt.contains(j2)) {
continue;}
else {
double c1=0;
double c2=0;
double c3=0;
if (ordersCopy.get(j2).getAssign()==2) {
c1=DistanceCaldis(scenario.get(j).getX(),scenario.get(j).getY(),
    ordersCopy.get(j2-1).getX(),ordersCopy.get(j2-1).getY());
c2=DistanceCaldis(scenario.get(j).getX(),scenario.get(j).getY(),
    ordersCopy.get(j2+1).getX(),ordersCopy.get(j2+1).getY());
c3=DistanceCaldis(ordersCopy.get(j2+1).getX(),ordersCopy.get(j2+1).getY
    (),ordersCopy.get(j2-1).getX(),ordersCopy.get(j2-1).getY());
}
else if (ordersCopy.get(j2+1).getAssign()==2){
c1=DistanceCaldis(scenario.get(j).getX(),scenario.get(j).getY(),
    ordersCopy.get(j2).getX(),ordersCopy.get(j2).getY());
c2=DistanceCaldis(scenario.get(j).getX(),scenario.get(j).getY(),
    ordersCopy.get(j2+2).getX(),ordersCopy.get(j2+2).getY());
c3=DistanceCaldis(ordersCopy.get(j2).getX(),ordersCopy.get(j2).getY(),
    ordersCopy.get(j2+2).getX(),ordersCopy.get(j2+2).getY());
}
else {
c1=DistanceCaldis(scenario.get(j).getX(),scenario.get(j).getY(),
```

```
    ordersCopy.get(j2).getX(),ordersCopy.get(j2).getY());
c2=DistanceCaldis(scenario.get(j).getX(),scenario.get(j).getY(),
    ordersCopy.get(j2+1).getX(),ordersCopy.get(j2+1).getY());
c3=DistanceCaldis(ordersCopy.get(j2+1).getX(),ordersCopy.get(j2+1).getY
    (),ordersCopy.get(j2).getX(),ordersCopy.get(j2).getY());
}
double cost=c1+c2-c3;
if (cost<=min) {
min=cost;
indexofreq=j;
indexofint=j2+1;
}}}}
ordersCopy.add(indexofint, scenario.get(indexofreq));
//Checking feasibility of the tour for the selected request
if (feasibility(ordersCopy, indexofint)==true) {
scenario.remove(indexofreq);
forbInt.clear();
a++;
loop=0;}
else {
ordersCopy.remove(indexofint);
forbInt.add(indexofint-1);
loop++;}
if (loop>ordersCopy.size() || scenario.size()==0) {
break;} }
//Assigning times to requests
int timeInt=-1;
for (int i = 0; i < ordersCopy.size()-1; i++) {
int distance=DistanceCal(ordersCopy.get(i+1).getX(),ordersCopy.get(i+1)
    .getY(), ordersCopy.get(i).getX(), ordersCopy.get(i).getY());
timeInt+=distance;
if (ordersCopy.get(i).getAssign()==2) {
timeInt-=distance;
timeInt=ordersCopy.get(i).getEnd();
continue;
}
```

```
else if (ordersCopy.get(i+1).getAssign()==1 ) {
timeInt=ordersCopy.get(i+1).getEnd();
continue;}
else if (ordersCopy.get(i+1).getAssign()==2) {
timeInt-=distance;
timeInt=ordersCopy.get(i+1).getEnd();
continue;}
else{
ordersCopy.get(i+1).setStart(timeInt);
ordersCopy.get(i+1).setEnd(timeInt+(Request.PatService/Request.TimeSlot
    ));
timeInt+=(Request.PatService/Request.TimeSlot);}}
//Find if the patient is assigned, finding which time slot he or she is
    assigned
for (int i = 0; i < ordersCopy.size(); i++) {
if (ordersCopy.get(i).getNumber()==re.getNumber()){
counter++;
timeSlot.add(ordersCopy.get(i).getStart());}}}
//Calculate Frequency
int maxFre=Integer.MIN_VALUE;
int slot=0;
for (int i = 0; i < timeSlot.size(); i++) {
int a=Collections.frequency(timeSlot, timeSlot.get(i));
if (a>=maxFre) {
maxFre=a;
slot=timeSlot.get(i);}}
int[] results=new int[2];
results[0]=counter;
results[1]=(slot);
return results;
}
```